

PARTICLE FILTERING IN HIGH-DIMENSIONAL CHAOTIC SYSTEMS

NISHANTH LINGALA¹, N. SRI NAMACHCHIVAYA¹, NICOLAS PERKOWSKI²,
AND HOONG C. YEONG¹

ABSTRACT. We present an efficient particle filtering algorithm for multiscale systems, that is adapted for simple atmospheric dynamics models which are inherently chaotic. Particle filters represent the posterior conditional distribution of the state variables by a collection of particles, which evolves and adapts recursively as new information becomes available. The difference between the estimated state and the true state of the system constitutes the error in specifying or forecasting the state, which is amplified in chaotic systems that have a number of positive Lyapunov exponents. The purpose of the present paper is to show that the homogenization method developed in Imkeller et al. [2012], which is applicable to high dimensional multi-scale filtering problems, along with important sampling and control methods can be used as a basic and flexible tool for the construction of the proposal density inherent in particle filtering. Finally, we apply the general homogenized particle filtering algorithm developed here to the Lorenz'96 (Lorenz [1995]) atmospheric model that mimics mid-latitude atmospheric dynamics with microscopic convective processes.

1. INTRODUCTION

The main goal of filtering is to obtain, recursively in time, the best statistical estimate of a natural or physical system based on noisy partial observations of the same. More precisely, filtering problems consist of an unobservable signal process and an observation process that is a function of the signal corrupted by noise. This is given by the conditional distribution of the signal given the observation process. This paper deals with real time filtering of chaotic signals from atmospheric models involving many degrees of freedom. Since small errors in our estimate of the current state of a chaotic system can grow to have a major impact on the subsequent forecast, better estimate of the signal is needed to make accurate predictions of the future state. This is obviously a problem with significant importance in real time filtering and prediction of weather and climate which involves coupled atmosphere-ocean system as well as the spread of hazardous plumes or pollutants governed by extremely complex flows. A complete study of these complex systems with practical impact, involves models of extremely unstable, chaotic dynamical systems with several million degrees of freedom.

Proper and accurate climate models can only be obtained by combining the models with data. Lower dimensional climate models require first the identification of slowly evolving climate modes and fast evolving non-climate modes. Contrary to standard initial value problems, we do not have access to the initial state of these dynamical systems. Instead, we have a known initial probability density function for the initial state and treat the model states as realizations of a random variable. Lower dimensional climate models also contain noise terms that account for the interaction of the resolved climate modes with the neglected non-climate modes. Hence these problems require efficient new algorithms to estimate the present and future state of the climate models, based upon corrupted, distorted, and possibly partial observations of the climate modes and fast evolving non-climate modes. While perfect determination of the state is impossible under these noisy observations, it may still be desirable to obtain probabilistic estimates of the state conditioned on the information that is available.

The sheer number of calculations required in directly solving large-scale random dynamical systems becomes computationally overwhelming. Hence, we consider the Lorenz'96 model of

Key words and phrases. particle filtering in high dimensions; chaotic systems; Lorenz'96; nonlinear filtering; dimensional reduction; homogenization.

type II (Lorenz [1995]) with two time-scale simplified ordinary differential equation describing advection, damping and forcing of some (slow) resolved atmospheric variables being coupled to some (fast) sub-scale variables as a nontrivial example of an atmospheric dynamics model. Though the model considered in this paper is a simple chaotic “toy model” of the atmosphere, which is an excellent test-bed for schemes that will be developed in Section 4, it has a more realistic dimension than the multitude of “Lorenz 1963”-type studies (i.e., 360 dimensions rather than 3 dimensions). Even though this simple model is still a long way from the Primitive Equations that we need to eventually study, it is hoped that the filtering methods presented here can be ultimately adapted for the realistic models that need to be computationally solved for weather and climate predictions.

In the past decade, vast amount of data has been produced via various sources from network-connected sensor arrays to satellites, all with a wide range of scale separations. To deal with such an onslaught of data, it is necessary to have a new framework capable of harnessing and processing these data with multiscale models. Hence, a data assimilation scheme that can handle chaotic systems that are sensitive to initial conditions (characterized by positive Lyapunov exponents) and noisy multiscale observations is needed. The theory of nonlinear filtering forms the framework in our study for the assimilation of data into multiscale models.

Particle filters represent the posterior conditional distribution of the state variables by a collection of particles, which evolves and adapts recursively as new information becomes available. Hence, particle filtering provides a recursive procedure for estimating an evolving state from a noisy observation process. Despite the general applicability and rigorous convergence results, particle filters have not been extensively used in state estimation of high dimensional problems, for example, weather prediction, as pointed out in Snyder et al. [2008]. This is due to the fact that a large number of particles is required and the particle filters suffer from particle degeneracy (see, for example, Daum and Huang [2003], Snyder et al. [2008]) in high dimensional systems. In this paper, we combine our study of stochastic dimensional reduction and nonlinear filtering to provide a rigorous framework for developing an algorithm for computing lower dimensional particle filters which are specifically adapted to the complexities of the underlying multiscale signal. When the rates of change of different model variables differ by orders of magnitude, efficient data assimilation can be accomplished by constructing a particle filter approach for nonlinear filtering equations for the coarse-grained signal. In moderate dimensional problems, particle filters are an attractive alternative to numerical approximation of the stochastic partial differential equations (SPDEs) by finite difference or finite element methods. Our approach described in this paper, not only reduces the computational burden for real time applications but also helps solve the problem of particle degeneracy. First objective is to predict the self-contained description of the coarse-grained dynamics without fully resolving the dynamics described in fast scales. In these problems, extracting coarse-grained dynamics is at the heart a problem of weak convergence of stochastic processes, or more precisely weak convergence of the laws of Markov processes.

We begin in Section 2 by presenting the general formulation of the multiscale nonlinear filtering problem. Here we introduce the homogenized equations that were derived in Imkeller et al. [2012] for the reduced dimension unnormalized filter. In Section 3 we present the model that was originally suggested by Lorenz (Lorenz’96), which incorporated a pattern with convective scales and introduced a crude model of K slow variables plus JK fast variables that varies with two distinct time scales. This is an excellent test-bed for data assimilation schemes that are developed in this paper. Section 4 outlines a sequential particle filtering algorithm (the Sequential Importance Sampling method) and a numerical algorithm for dimensional reduction (the Heterogeneous Multiscale Method of Vanden-Eijnden [2003], E et al. [2005]). These two methods are combined to give an efficient algorithm for particle filtering in a multiscale environment. Finally, we present in Section 5, the results from several data assimilation experiments on the Lorenz’96 model and discuss future research directions based on “adaptive” or “targeted” observations and sensor placement.

2. FORMULATION OF MULTISCALE NONLINEAR FILTERING PROBLEMS

The results presented here are set within the context of slow-fast dynamical systems, where the rates of change of different variables differ by orders of magnitude. The effects of the multiscale signal and observation processes via the study of lower dimensional Zakai equations in a canonical way was presented in Imkeller et al. [2011, 2012], where the convergence of the optimal filter to the homogenized filter is shown using backward stochastic differential equations (BSDEs) and asymptotic techniques. This paper provided rigorous mathematical results that support the numerical algorithms based on the idea that stochastically averaged models provide qualitatively useful results which are potentially helpful in developing inexpensive lower-dimensional filtering. For the reduced nonlinear model an appropriate form of particle filter can be a viable and useful scheme. Hence, we present the numerical solution of the lower dimensional stochastic partial differential equation derived here, as it is applied to several higher dimensional multiscale applications.

We will present the main result of Imkeller et al. [2012] here. Let $(\Omega, \mathcal{F}, (\mathcal{F}_t), \mathbb{Q})$ be a filtered probability space that supports a standard Brownian motion $(V, W, B) \in \mathbb{R}^{k \times l \times d}$. We consider the signal process that is the solution of the two time scale stochastic differential equations (SDEs) driven by (W, V) :

$$(1a) \quad dX_t^\varepsilon = b(X_t^\varepsilon, Z_t^\varepsilon)dt + \sigma(X_t^\varepsilon, Z_t^\varepsilon)dV_t, \quad X_0^\varepsilon = \xi \in \mathbb{R}^m$$

$$(1b) \quad dZ_t^\varepsilon = \frac{1}{\varepsilon}f(X_t^\varepsilon, Z_t^\varepsilon)dt + \frac{1}{\sqrt{\varepsilon}}g(X_t^\varepsilon, Z_t^\varepsilon)dW_t, \quad Z_0^\varepsilon = \eta \in \mathbb{R}^n.$$

Here, $\varepsilon \ll 1$ is the timescale separation parameter, so X^ε is the slow component and Z^ε is the fast component. W , V and B are independent of each other as well as the random initial conditions ξ and η . The functions f , g , b , σ are assumed to be Borel-measurable.

Associated to the signal is the d -dimensional observation Y^ε that is perturbed by the Brownian motion B , given by

$$Y_t^\varepsilon = \int_0^t h(X_s^\varepsilon, Z_s^\varepsilon)ds + B_t,$$

where h is a Borel-measurable function. Define the sigma algebra generated by the observation, $\mathcal{Y}_t^\varepsilon = \sigma(Y_s^\varepsilon : 0 \leq s \leq t) \vee \mathcal{N}$, where \mathcal{N} are the \mathbb{Q} -negligible sets.

The main objective of filtering theory is to obtain the best estimate of the signal process based on information from the observation. The best estimate, called the optimal filter (the Zakai equation), is a conditional expectation that satisfies a recursive equation driven by the observation (Zakai [1969]). For a formal definition, denote the optimal filter for the multiscale system by π^ε , a finite measure on \mathbb{R}^{m+n} . The goal is to calculate the (normalized) observation-dependent filter $\pi_t^\varepsilon(\varphi) \stackrel{\text{def}}{=} \mathbb{E}_{\mathbb{Q}}[\varphi(X_t^\varepsilon, Z_t^\varepsilon)|\mathcal{Y}_t^\varepsilon]$, where φ is a bounded measurable function on \mathbb{R}^{m+n} .

In order to calculate the specified conditional expectation, it is easier to work on a new probability space \mathbb{P}^ε , on which the observation Y^ε is a Brownian motion independent of the signal noise (W, V) . \mathbb{P}^ε is related to the original probability space \mathbb{Q} by the Girsanov transform Bain and Crisan [2009]

$$(2) \quad D_t^\varepsilon \stackrel{\text{def}}{=} \frac{d\mathbb{P}^\varepsilon}{d\mathbb{Q}} \Big|_{\mathcal{F}_t} = \exp \left(- \int_0^t h(X_s^\varepsilon, Z_s^\varepsilon)^T dB_s - \frac{1}{2} \int_0^t |h(X_s^\varepsilon, Z_s^\varepsilon)|^2 ds \right),$$

which effectively removes the drift h from the observation equation. Define an unnormalized filter as $\rho_t^\varepsilon(\varphi) \stackrel{\text{def}}{=} \mathbb{E}_{\mathbb{P}^\varepsilon}[\varphi(X_t^\varepsilon, Z_t^\varepsilon)(D_t^\varepsilon)^{-1}|\mathcal{Y}_t^\varepsilon]$. The normalized and unnormalized filters are related using the Girsanov transform:

$$\pi_t^\varepsilon(\varphi) = \mathbb{E}_{\mathbb{Q}}[\varphi(X_t^\varepsilon, Z_t^\varepsilon)|\mathcal{Y}_t^\varepsilon] = \frac{\mathbb{E}_{\mathbb{P}^\varepsilon}[\varphi(X_t^\varepsilon, Z_t^\varepsilon)(D_t^\varepsilon)^{-1}|\mathcal{Y}_t^\varepsilon]}{\mathbb{E}_{\mathbb{P}^\varepsilon}[(D_t^\varepsilon)^{-1}|\mathcal{Y}_t^\varepsilon]} = \frac{\rho_t^\varepsilon(\varphi)}{\rho_t^\varepsilon(1)}.$$

The unnormalized filter ρ^ε satisfies the Zakai equation (see, for example, Bain and Crisan [2009]):

$$d\rho_t^\varepsilon(\varphi) = \rho_t^\varepsilon(\mathcal{L}^\varepsilon \varphi)dt + \rho_t^\varepsilon(h\varphi)dY_t^\varepsilon, \quad \rho_0^\varepsilon(\varphi) = \mathbb{E}_\mathbb{Q}[\varphi(X_0^\varepsilon, Z_0^\varepsilon)].$$

Here, $\mathcal{L}^\varepsilon = \frac{1}{\varepsilon}\mathcal{L}_F + \mathcal{L}_S$ is the differential operator associated to $(X^\varepsilon, Z^\varepsilon)$, with

$$\begin{aligned} \mathcal{L}_F &= \sum_{i=1}^n f_i(x, z) \frac{\partial}{\partial z_i} + \frac{1}{2} \sum_{i,j=1}^n (gg^T)_{ij}(x, z) \frac{\partial^2}{\partial z_i \partial z_j} \\ \mathcal{L}_S &= \sum_{i=1}^m b_i(x, z) \frac{\partial}{\partial x_i} + \frac{1}{2} \sum_{i,j=1}^m (\sigma\sigma^T)_{ij}(x, z) \frac{\partial^2}{\partial x_i \partial x_j} \end{aligned}$$

where \cdot^T denotes the transpose of a matrix or a vector.

Next, we briefly present a result from stochastic averaging theory for the multiscale setting of the problem considered. We assume that for every $x \in \mathbb{R}^m$, the solution Z^x of (1b) with $X^\varepsilon = x$ fixed is ergodic and converges rapidly to its unique stationary distribution $p_\infty(x : \cdot)$. In this case, it is well known that X^ε converges in distribution to a diffusion X^0 governed by an SDE

$$dX_t^0 = \bar{b}(X_t^0)dt + \bar{\sigma}(X_t^0)dV_t$$

for appropriately averaged \bar{b} and $\bar{\sigma}$. In other words, a stochastically averaged model provides a qualitatively useful approximation to the actual multiscale system. Hence, if we are only interested in estimating the *slow* process, or the “coarse-grained” dynamics, then we should make use of the homogenized diffusion X^0 . Specifically, if we are only interested in the x -marginal, $\pi^{\varepsilon,x}$, of the optimal π^ε , then this X^0 can be used to construct an averaged, or homogenized, filter π^0 that can appropriately replace $\pi^{\varepsilon,x}$. In high-dimensional problems, π^0 would be preferable to $\pi^{\varepsilon,x}$ since calculation of π^0 using X^0 does not directly involve calculations of the fast process.

By making use of X^0 , we would like to find a homogenized (unnormalized) filter ρ^0 that satisfies

$$d\rho_t^0(\varphi) = \rho_t^0(\bar{\mathcal{L}}\varphi)dt + \rho_t^0(\bar{h}\varphi)dY_t^\varepsilon, \quad \rho_0^0(\varphi) = \mathbb{E}_\mathbb{Q}[\varphi(X_0^0)],$$

such that for small ε , the x -marginal of ρ^ε , $\rho^{\varepsilon,x}$, is close to ρ^0 . We let the generator $\bar{\mathcal{L}}$ of X^0 be defined as

$$\bar{\mathcal{L}} = \sum_{i=1}^m \bar{b}_i(x) \frac{\partial}{\partial x_i} + \frac{1}{2} \sum_{i,j=1}^m \bar{a}_{ij}(x, z) \frac{\partial^2}{\partial x_i \partial x_j}$$

where $\bar{b}(x) = \int b(x, z)p_\infty(x, dz)$ and $\bar{a} = \int (\sigma\sigma^T)(x, z)p_\infty(x, dz)$. Also, define $\bar{h}(x) = \int h(x, z)p_\infty(x, dz)$. Note that the homogenized filter is still driven by the real observation Y^ε and not by a “homogenized observation”, which is practical for implementation of the homogenized filter in applications since such averaged observation is usually not available. However, even if such homogenized observation is available, using it would lead to loss of information for estimating the signal compared to using the actual observation.

Now define the measure-valued processes π^0 and $\pi^{\varepsilon,x}$ in terms of ρ^0 and $\rho^{\varepsilon,x}$ as π^ε was in terms of ρ^ε :

$$\pi_t^0(\varphi) = \frac{\rho_t^0(\varphi)}{\rho_t^0(1)} \quad \text{and} \quad \pi_t^{\varepsilon,x}(\varphi) = \frac{\rho_t^{\varepsilon,x}(\varphi)}{\rho_t^{\varepsilon,x}(1)}.$$

The main result of Imkeller et al. [2012] is that under appropriate assumptions on the coefficients of (1a), (1b) and h , there exists a metric d on the space of probability measures such that for every $T \geq 0$ there exists $C > 0$ such that

$$\mathbb{E}_\mathbb{Q} [d(\pi_T^{\varepsilon,x}, \pi_T^0)] \leq \sqrt{\varepsilon}C, \text{ i.e. } \lim_{\varepsilon \rightarrow 0} \mathbb{E}_\mathbb{Q} [d(\pi_T^{\varepsilon,x}, \pi_T^0)] = 0 \text{ for any } T > 0.$$

In other words, the x -marginal of the optimal filter π^ε converges to the averaged filter π^0 in the space of probability measures as the separation parameter ε goes to 0. Hence, the homogenized filter is an appropriate measure to use in place of the actual $\pi^{\varepsilon,x}$ in estimating the “coarse-grained” dynamics X^ε in a setting with wide timescale separation. In terms of filtering applications, π^0 presents the advantage of not requiring exact knowledge of the fast dynamics for the purpose of estimating the “coarse-grained” dynamics. Only knowledge of the invariant measure of Z^x is required, hence by applying appropriate multiscale averaging numerical schemes, computation and information storage for the fast dynamics can be reduced.

The convergence of the normalized filters, $\pi^{\varepsilon,x} \rightarrow \pi^0$, was shown by first obtaining the convergence of the unnormalized filters, $\rho^{\varepsilon,x}$ to ρ^0 . The dual representations of $\rho_T^{\varepsilon,x}(\varphi)$ and $\rho_T^0(\varphi)$ were introduced as in Pardoux [1979]:

$$v_t^{\varepsilon,T,\varphi}(x,z) \stackrel{\text{def}}{=} \mathbb{E}_{\mathbb{P}_{t,x,z}^\varepsilon}[\varphi(X_T^\varepsilon) (D_{t,T}^\varepsilon)^{-1} | \mathcal{Y}_{t,T}^\varepsilon], \quad \text{and} \quad v_t^{0,T,\varphi}(x) = \mathbb{E}_{\mathbb{P}_{t,x}^0}[\varphi(X_T^0) (D_{t,T}^0)^{-1} | \mathcal{Y}_{t,T}^\varepsilon].$$

$\mathbb{P}_{t,x,z}^\varepsilon$ and $\mathbb{P}_{t,x}^0$ are the respective measures under which $(X^\varepsilon, Z^\varepsilon)$ and X^0 are governed by the same dynamics as under \mathbb{P}^ε and P^0 , but $(X^\varepsilon, Z^\varepsilon)$ and X^0 stays in (x, z) and x until time t . $D_{t,T}^\varepsilon$ and $D_{t,T}^0$ were defined as the Girsanov transform (2), but with moving limit of integration t ,

$$D_t^\varepsilon \stackrel{\text{def}}{=} \frac{d\mathbb{P}^\varepsilon}{dQ} \Big|_{\mathcal{F}_t} = \exp \left(- \int_t^T h(X_s^\varepsilon, Z_s^\varepsilon)^T dB_s - \frac{1}{2} \int_t^T |h(X_s^\varepsilon, Z_s^\varepsilon)|^2 ds \right),$$

$$D_{t,T}^0 = \exp \left(- \int_t^T \bar{h}(X_r^0)^T dY_r^\varepsilon + \frac{1}{2} \int_t^T |\bar{h}(X_r^0)|^2 dr \right),$$

and $\mathcal{Y}_{t,T}^\varepsilon = \sigma(Y_r^\varepsilon - Y_t^\varepsilon : t \leq r \leq T) \vee \mathcal{N}$ is the filtration generated by the observation over $[t, T]$, minus observation history from 0 to t . The Markov property of $(X^\varepsilon, Z^\varepsilon, X^0)$ results gives the relations between the unnormalized filters and the respective duals:

$$(3) \quad \rho_T^{\varepsilon,x}(\varphi) = \int v_0^{\varepsilon,T,\varphi}(x,z) \mathbb{Q}_{(X_0^\varepsilon, Z_0^\varepsilon)}(dx, dz) \quad \text{and} \quad \rho_T^0(\varphi) = \int v_0^{0,T,\varphi}(x) \mathbb{Q}_{X_0^0}(dx).$$

Note that $\mathbb{Q}_{X_0^0} = \mathbb{Q}_{X_0^\varepsilon}$, because the homogenized process has the same starting distribution as the un-homogenized one.

For fixed T and $\varphi \in C_b^2(\mathbb{R}^m, \mathbb{R})$, we will write $v_t^\varepsilon = v_t^{\varepsilon,T,\varphi}$ and $v_t^0 = v_t^{0,T,\varphi}$. The dual process v_t^ε essentially represents the conditional expectation $\rho_T^{\varepsilon,x}$ by an alternate conditional expectation that is run backwards in time from T to 0. For fixed starting point (x, z) at $t = 0$, this backward in time conditional expectation is constructed using processes $(X_T^\varepsilon, Z_T^\varepsilon)$ that started at (x, z) and ran backwards to $t = 0$. By integrating v_0^ε over $\mathbb{Q}_{(X_0^\varepsilon, Z_0^\varepsilon)}$, we are integrating over all possible starting points (x, z) , hence giving ρ_T^ε . This interpretation is similar for v^0 of ρ^0 .

From (3), we have

$$(4) \quad \mathbb{E}[|\rho_T^{\varepsilon,x}(\varphi) - \rho_T^0(\varphi)|^p] \leq \int \mathbb{E}[|v_0^\varepsilon(x, z) - v_0^0(x)|^p] \mathbb{Q}_{(X_0^\varepsilon, Z_0^\varepsilon)}(dx, dz).$$

So, if $|v_0^\varepsilon(x, z) - v_0^0(x)|$ is small, then $|\rho_T^{\varepsilon,x}(\varphi) - \rho_T^0(\varphi)|$ will also be small as long as $\mathbb{Q}_{(X_0^\varepsilon, Z_0^\varepsilon)}$ is well behaved and (4) will lead to the normalized filter convergence result. Therefore, the aim is to show that for nice test functions φ , $\mathbb{E}[|v_0^\varepsilon(x, z) - v_0^0(x)|^p]$ is small.

The key point is that v^ε and v^0 solve backward SPDEs. We formally expand v^ε as

$$v_t^\varepsilon(x, z) = \underbrace{u_t^0(x, z)}_{v^0(t,x)} + \underbrace{\varepsilon u_{t/\varepsilon}^1(x, z)}_{\psi(t,x,z)} + \underbrace{\varepsilon^2 u_{t/\varepsilon}^2(x, z)}_{R(t,x,z)}.$$

Then, v^0 , ψ and R satisfy linear partial differential equations with appropriate terminal conditions. By existence and uniqueness of the solutions to these *linear* equations, we can apply superposition to obtain that indeed

$$v_t^\varepsilon(x, z) = v_t^0(x) + \psi_t(x, z) + R_t(x, z),$$

where ψ and R are the corrector and remainder terms, respectively. Based on the expansion, the problem of showing L^p -convergence of v^ε to v^0 reduces to showing L^p -convergence of ψ and R to 0.

Details of the proof are provided in Imkeller et al. [2012]. The outline of the method of proof is as follows: The backwards SPDEs were converted to their respective probabilistic representations, which are backward doubly stochastic differential equations (BDSDEs). The diffusion operators were replaced by the associated diffusions and explicit estimates for the finite dimensional BDSDEs in terms of the transition density function of the fast process were able to be obtained. Pardoux and Veretennikov [2003] proved very precise estimates for this transition function, and were used to obtain the desired bounds on ψ and R .

3. THE LORENZ'96 SYSTEM

The systematic strategy for the identification of slowly evolving climate modes and fast evolving non-climate modes requires a lengthy analysis. In this paper, we consider a simple atmospheric model, which nonetheless exhibits many of the difficulties arising in realistic models, to gain insight into predictability and data assimilation. The Lorenz'96 model was originally introduced (in Lorenz [1995]) to mimic multiscale mid-latitude weather, considering an unspecified scalar meteorological quantity at K equidistant grid points along a latitude circle:

$$(5) \quad \dot{X}_t^k = -X_t^{k-1}(X_t^{k-2} - X_t^{k+1}) - X_t^k + F_x + \frac{h_x}{J} \sum_{j=1}^J Z_t^{k,j}, \quad k = 1, \dots, K,$$

$$(6) \quad \dot{Z}_t^{k,j} = \frac{1}{\varepsilon} \left\{ -Z_t^{k,j+1}(Z_t^{k,j+2} - Z_t^{k,j-1}) - Z_t^{k,j} + h_z X_t^k \right\}, \quad j = 1, \dots, J.$$

Equation (5) describes the dynamics of some atmospheric quantity X , and X_t^k can represent the value of this variable at time t , in the k^{th} sector defined over a latitude circle in the mid-latitude region (*Note:* We use superscripts k and j to conform with the typical spatial indexing notation used for the Lorenz '96 model. In sections that follow, subscripts k and j will be used as discrete time indices, not to be confused with the spatial indices of the Lorenz model). The latitude circle is divided into K sectors (with values of K ranging from $K = 4$ to $K = 36$). Each X_t^k is coupled to its neighbors X_t^{k+1} , X_t^{k-1} , and X_t^{k-2} by (5). (5) applies for all values of k by letting $X_t^{k+K} = X_t^{k-K} = X_t^k$, so, for example for $k = 1$, $X_t^{k+1} = X_t^2$, $X_t^{k-1} = X_t^{36}$, and $X_t^{k-2} = X_t^{35}$.

The system was extended to study the influence of multiple spatio-temporal scales on the predictability of atmospheric flow by the dividing each segment k into J subsectors ($J = 4$ to $J = 10$), and introducing a fast variable, $Z_t^{k,j}$ given by (6), associated with each subsector. Thus, each X_t^k represents a slowly-varying, large amplitude atmospheric quantity, with J fast-varying, low amplitude, similarly coupled quantities, $Z_t^{k,j}$, associated with it. In the context of climate modeling, the slow component is also known as the resolved climate modes while the rapidly-varying component is known as the unresolved non-climate modes. The coupling terms between neighbors model advection between sectors and subsectors, while the coupling between each sector and its subsectors model damping within the model. The model is subjected to linear external forcing, F_x , on the slow timescale.

The dynamics of the unresolved modes can be modified to include nonlinear self-interaction effects by adding forcing in the form of stochastic terms (see, for example, Majda et al. [2001, 2003]). The use of stochastic terms to represent nonlinear self-interaction effects at short timescales in the unresolved modes is appropriate if we are only interested in the coarse-grained dynamics occurring in the long timescale. This is called stochastic consistency in Majda et al. [2003]. Considering (6), where only quadratic nonlinearity is present, the motivation behind adding stochastic forcing is thus to model higher order self-interaction effects. This can be done using a mean-zero Ornstein-Uhlenback process, specifically a process with amplitude $\frac{1}{\sqrt{\varepsilon}}$, that

is,

$$(7) \quad \dot{Z}_t^{k,j} = \frac{1}{\varepsilon} \left\{ -Z_t^{k,j+1}(Z_t^{k,j+2} - Z_t^{k,j-1}) - Z_t^{k,j} + h_z X_t^k \right\} + \frac{1}{\sqrt{\varepsilon}} \zeta(t).$$

This is done more explicitly in a general context in Section 4. The two-scale Lorenz'96 model was also used extensively by Wilks [2005] to study stochastic parametrization and by Lorenz and Emanuel [1998] for analyzing targeted observations, and by Herrera et al. [2011] and several others for analyzing the influence of large-scale spatial patterns on the growth of small perturbations.

4. HOMOGENIZED HYBRID PARTICLE FILTER (HHPF)

Numerical simulation of multiscale dynamical systems is quite problematic because of the wide separation in the time-scales involved. Based on the results presented in Section 2, we combine our study of stochastic dimensional reduction and nonlinear filtering to provide a rigorous framework for developing an algorithm for lower dimensional particle filters specifically adapted to the complexities of underlying multiscale signals. The proposed particle filter algorithm is adapted to a system with time-scales separation by incorporating a homogenization scheme in the nonlinear filter, in conjunction with the stochastic dimensional reduction results of Section 2.

In order to apply stochastic dimensional reduction in the problem presented in Section 3, we introduce a stochastic forcing term in (6) to represent external forcing effects on the system due to unresolved dynamics. To illustrate the homogenization scheme, we consider a general system of stochastic differential equations (SDEs) that corresponds with the problem of Section 3, with stochastic forcing in the fast component:

$$(8) \quad \begin{aligned} \dot{X}_t^\varepsilon &= b(X_t^\varepsilon, Z_t^\varepsilon), \quad X_0^\varepsilon \sim \mathcal{N}(\mu_0^x, \sigma_0^x) \\ \dot{Z}_t^\varepsilon &= \varepsilon^{-1} f(X_t^\varepsilon, Z_t^\varepsilon) + \varepsilon^{-1/2} g(X_t^\varepsilon, Z_t^\varepsilon) \dot{W}_t, \quad Z_0^\varepsilon \sim \mathcal{N}(\mu_0^z, \sigma_0^z). \end{aligned}$$

In the following two subsections we explain the Heterogeneous Multiscale Method and Sequential Importance Sampling. These techniques are then combined in the Homogenized Hybrid Particle Filter.

4.1. Multiscale numerical integration. For numerical simulations of (8), the timestep δt required for the forward integration of the fast process Z^ε needs to be smaller than ε for stability of the integration scheme. However, with such timestep, significant changes in the slow variable can only be seen on the time-scale of $\mathcal{O}(1)$, i.e., much of the computational resources is wasted in solving for an almost stationary evolution of the slow process X^ε .

Based on the results presented in Section 2, we can solve this problem by adopting a reduced system through stochastic averaging, that is,

$$(9) \quad \dot{\bar{X}}_t = \bar{b}(\bar{X}_t),$$

where

$$(10) \quad \bar{b}(x) = \int b(x, z) \mu_x(dz).$$

By adopting (9) with (10), we are under the assumption that (8) is exponentially mixing, and the Doeblin condition is satisfied. Stated informally: With the slow process fixed at $X_t^\varepsilon = x$, the corresponding fast process Z^x has a unique invariant measure $\mu_x(dz)$, i.e. the transition probability measure $P^x(z, t)$ of Z_t^x converges exponentially to $\mu_x(dz)$ in the weak sense as $t \rightarrow \infty$, locally uniformly in x and z . This implies that for any test function $\varphi \in C_b(\mathbb{R}^m)$

$$\mathbb{E}_z[\varphi(Z_t^x)] \rightarrow \int \varphi(z) \mu_x(dz) \quad \text{as } t \rightarrow \infty,$$

uniformly in x and z in any compact set.

However, the evaluation of the high dimensional integration in (10) is nontrivial, and usually it is impossible to obtain an invariant distribution of the fast variable analytically. To determine the invariant distribution numerically, we adopt the Heterogeneous Multiscale Method (HMM)

introduced in Vanden-Eijnden [2003]. The HMM is a method of determining the effective dynamics (9) through numerical approximation of the invariant distribution of the fast process. It is based on the observation that the fast variable Z^ε reaches its invariant distribution (equilibrium) on a time-scale much smaller than the time-scale needed to evolve the slow variable X^ε (the Doeblin condition). This implies that we could use a much larger timestep value Δt for the evolution of the slow process while keeping the timestep for the fast process small for stability.

Following the procedure presented in Vanden-Eijnden [2003], we describe the HMM. The evolution of the averaged equation (9) is approximated numerically using a forward integration scheme. For simplicity, we will use the Euler and Euler-Maruyama schemes in the following description, although higher order schemes are also applicable (see, for example, Kloeden and Platen [1992]). Consider the interval $[0, T]$ discretized into timesteps of size $\Delta t = \lfloor \frac{T}{N} \rfloor$. Let $t_k \stackrel{\text{def}}{=} k\Delta t$ and write X_{t_k} as X_k .

$$(11) \quad \bar{X}_{k+1} = \bar{X}_k + \tilde{b}(\bar{X}_k)\Delta t$$

is the *macro-solver* with Δt being the *macro-timestep* (Note: As mentioned in Section 3, subscript k here indicates time index, different from the spatial index superscript k in the Lorenz '96 model). The value of \tilde{b} , which is an approximation of the averaged coefficient \bar{b} in (10), can be calculated as

$$(12) \quad \tilde{b}(\bar{X}_k) = \frac{1}{MN_m} \sum_{r=1}^M \sum_{j=n_T}^{n_T+N_m} b(\bar{X}_k, Z_{k,j}^{\varepsilon,r})$$

where M is the number of replicas of the fast process Z for spatial averaging, N_m is the number of *micro-timesteps* δt for time averaging, and n_T is the number of micro-timesteps skipped to eliminate transient effects. The evolution of $Z_{k,j}^{\varepsilon,r}$ is governed by the following *micro-solver*

$$(13) \quad Z_{k,j+1}^{\varepsilon,r} = Z_{k,j}^{\varepsilon,r} + \frac{1}{\varepsilon} f(\bar{X}_k, Z_{k,j}^{\varepsilon,r})\delta t + \frac{1}{\sqrt{\varepsilon}} g(\bar{X}_k, Z_{k,j}^{\varepsilon,r})\delta W.$$

Note that a combination of spatial and temporal averaging is used in (12) but it is shown in Vanden-Eijnden [2003] that the combinations of M , N_m , and δt can be chosen based on error analysis such that no spatial averaging or no spatial and temporal averaging is required. For more detailed explanation and error analysis, refer to the references Vanden-Eijnden [2003], Fatkullin and Vanden-Eijnden [2004].

4.2. Homogenized Hybrid Particle Filter (HHPF). The algorithm for the continuous-time HHPF is presented in Park et al. [2010]. As in standard particle filtering methods, the continuous-time equations can be discretized and the filtering can be done on the resulting discrete-time models. Here, we present the discrete-time version of the HHPF using the sequential importance sampling (SIS) algorithm, which is also commonly known as bootstrap filtering (see, for example, Arulampalam et al. [2002], Gordon et al. [1993]).

We will first provide a brief overview of the idea behind importance sampling and then illustrate how it is applied sequentially in particle filters. Then we present how the HHPF uses the SIS algorithm.

4.2.1. Importance Sampling. In particle filtering, there is always the need to represent some distribution using a collection of particles. When it is difficult to sample from a given distribution, the idea is to sample from another distribution that is more tractable to sample from, and properly use that sample to represent the distribution of interest.

Importance sampling is a technique for approximating integrals with respect to one probability distribution using a collection of samples from another. Let p be the target distribution of interest over space \mathbb{X} and $q \gg p$ (q is absolutely continuous with respect to p) be the distribution from which sampling is done (q is also called the proposal distribution). Denote by $\mathbb{E}_p[\cdot]$ and

$\mathbb{E}_q[\cdot]$ the expectation with respect to the distributions p and q , respectively. For any integrable function $\varphi : \mathbb{X} \rightarrow \mathbb{R}$, we have

$$(14) \quad \begin{aligned} \mathbb{E}_p[\varphi(X)] &= \int_{\mathbb{X}} \varphi(x) p(dx) = \int_{\mathbb{X}} \varphi(x) \frac{dp}{dq}(x) q(dx) \\ &= \int_{\mathbb{X}} \varphi(x) w(x) q(dx) = \mathbb{E}_q[w(X) \varphi(X)], \end{aligned}$$

where $w \stackrel{\text{def}}{=} \frac{dp}{dq}$.

A collection $\{x^i\}_{i=1}^{N_s}$ of N_s particles can be sampled from q and the particles can be weighted according to $w^i \propto \frac{dp}{dq}(x^i)$ to represent the target distribution p i.e.

$$p(x) \approx \sum_{i=1}^{N_s} w^i \delta(x - x^i).$$

The weights w^i are normalized such that $\sum w^i = 1$.

The strong law of large numbers can be employed to verify that the empirical average of φ with respect to the weighted sample from q converges as $N_s \rightarrow \infty$, with probability 1, to the expected value of φ under the target distribution p , i.e.

$$\int \left[\sum_{i=1}^{N_s} w^i \delta(x - x^i) \right] \varphi(x) dx = \sum_{i=1}^{N_s} w^i \varphi(x^i) \rightarrow \mathbb{E}_p[\varphi(X)].$$

4.2.2. Sequential Importance Sampling (SIS). The SIS algorithm is a technique of using Monte-Carlo simulations for Bayesian filtering through the incorporation of importance sampling. Consider a discrete-time signal X_{t_k} and a discrete-time observation Y_{t_k} and let $p(x_{0:k} | y_{0:k})$ be the density of the target posterior distribution at timestep t_k . The SIS algorithm approximates the target distribution using appropriately weighted samples from a proposal density $q(x_{0:k} | y_{0:k})$.

Suppose we represent $p(x_{0:k} | y_{0:k})$ using a collection $\{x_{0:k}^i\}_{i=1}^{N_s}$ of N_s particles sampled according to $q(x_{0:k} | y_{0:k})$ as

$$p(x_{0:k} | y_{0:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x - x_{0:k}^i),$$

where $w_k^i \propto \frac{p(x_{0:k}^i | y_{0:k})}{q(x_{0:k}^i | y_{0:k})}$, with $\sum_i w_k^i = 1$. We choose to arrive at proposal densities sequentially

$$q(x_{0:k} | y_{0:k}) = q(x_k | x_{k-1}, y_k) q(x_{0:k-1} | y_{0:k-1}).$$

Making use of the identity (see, for example, Arulampalam et al. [2002], equation (45))

$$p(x_{0:k} | y_{0:k}) = \frac{p(y_k | x_k) p(x_k | x_{k-1}) p(x_{0:k-1} | y_{0:k-1})}{p(y_k | y_{0:k-1})},$$

we write (after removing $p(y_k | y_{0:k-1})$ because it is same for all the particles)

$$(15) \quad \begin{aligned} w_k^i &\propto \frac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i) p(x_{0:k-1}^i | y_{0:k-1})}{q(x_k^i | x_{k-1}^i, y_k) q(x_{0:k-1}^i | y_{0:k-1})} \\ &\propto \frac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, y_k)} w_{k-1}^i. \end{aligned}$$

Hence we have a sequential formula for computing the unnormalized particle weights. Because our choice of q is done sequentially, the proposal density depends on the location of the particle only at the previous time step. Only x_{k-1}^i need be stored and the rest of the path $x_{0:k}^i$ can be discarded. We then have

$$p(x_k | y_{0:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x - x_k^i)$$

with weights updated according to (15) and normalized by $\sum_i w_k^i = 1$.

So, the SIS algorithm is this: At step $k - 1$ the location of particles $\{x_{k-1}^i\}_{i=1}^{N_s}$ is known. New observation y_k is recorded. Choose a form for the proposal $q(\cdot|x_{k-1}^i, y_k)$. Sample a particle according to this proposal density and get the new location x_k^i . Evaluate the number $q(x_k^i|x_{k-1}^i, y_k)$, and then evaluate the quantities in the numerator of equation (15), using the sensor dynamics and the signal dynamics. Then update the weights according to equation (15).

In filtering with a fixed number of particles, it is crucial to keep the variance of the weights to minimum: if a collection of particles is such that the weight is concentrated in a small number of particles, this collection represents the distribution poorly. One can better it by selecting a collection which has many particles near the region of high concentration and the particles sharing nearly equal weights. As can be seen from the weight update equation (15), the particle weights depend crucially on the choice of the proposal distribution q . So, q should be chosen with an aim of minimizing the variance of particle weights.

It can be shown Arulampalam et al. [2002] that the proposal density which keeps the variance of the weights to a minimum is

$$(16) \quad q^{\text{opt}}(x_k|x_{k-1}, y_k) = \frac{p(y_k|x_k)p(x_k|x_{k-1})}{\int p(y_k|x_k)p(x_k|x_{k-1})dx_k}.$$

Employing this in weight update equation 15 we have that, if we choose the optimal proposal density,

$$(17) \quad w_k^i \propto w_{k-1}^i \int p(y_k|x_k)p(x_k|x_{k-1}^i)dx_k.$$

In general case, it is difficult to sample from q^{opt} . However, it is easy when both the likelihood $p(y_k|x_k)$ and the conditional prior $p(x_k|x_{k-1})$ are Gaussians, cf. Arulampalam et al. [2002].

Consider the system dynamics and observation equation:

$$(18) \quad X_{k+1} = F(X_k) + \sigma_X W_{k+1}$$

$$(19) \quad Y_{k+1} = HX_{k+1} + \sigma_Y V_{k+1}$$

where W_k and V_k are independent centered Gaussian increments with variance one and $Q \stackrel{\text{def}}{=} \sigma_X \sigma_X^T$, $R \stackrel{\text{def}}{=} \sigma_Y \sigma_Y^T$ are strictly positive definite. Since this subsection on SIS and the next one on optimal control discuss a general method, we have used F to denote the vector field instead of b and f as in the previous sections. We then have that the likelihood $p(y_k|x_k) = \mathcal{N}(Hx_k, R)$ and the conditional prior $p(x_k|x_{k-1}) = \mathcal{N}(F(X_{k-1}), Q)$. Using (16), we have

$$(20) \quad \begin{aligned} q^{\text{opt}}(x_k|x_{k-1}, y_k) &= \mathcal{N}(F(x_{k-1}) + \alpha(x_{k-1}, y_k), \hat{Q}), \\ \hat{Q} &= (Q^{-1} + H^T R^{-1} H)^{-1}, \\ \alpha(x_{k-1}, y_k) &= \hat{Q} H^T R^{-1} (y_k - H F(x_{k-1})). \end{aligned}$$

This q^{opt} is a Gaussian. Once we have particle locations $\{x_{k-1}^i\}_{i=1}^{N_s}$ representing the posterior at time $k - 1$, and the observation y_k is recorded, we can evaluate $\alpha(x_{k-1}, y_k)$. We can then sample a particle x_k^i from the above Gaussian $\mathcal{N}(F(x_{k-1}^i) + \alpha(x_{k-1}^i, y_k), \hat{Q})$ and arrive at a collection of particles $\{x_k^i\}$. Alternatively, the particles can be evolved according to

$$(21) \quad X_{k+1} = F(X_k) + \alpha(X_k, y_{k+1}) + \hat{\sigma}_X W_{k+1}$$

where $\hat{\sigma}_X$ is such that $\hat{\sigma}_X \hat{\sigma}_X^T = \hat{Q}$. Then X_k^i behaves like a particle sampled from $q^{\text{opt}}(\cdot|x_{k-1}^i, y_k)$. The weights are updated according to (17): we will have

$$(22) \quad \begin{aligned} w_k^i &\propto w_{k-1}^i \exp \left\{ -\frac{1}{2} (y_k - H F(x_{k-1}^i))^T \hat{R}^{-1} (y_k - H F(x_{k-1}^i)) \right\}, \\ \hat{R}^{-1} &= R^{-1} (\mathbf{1} - H \hat{Q} H^T R^{-1}). \end{aligned}$$

4.2.3. *Stochastic optimal control approach.* We consider the same discrete-time nonlinear system with linear observation (18), (19). The particle method presented in this section consists of control terms in the “prognostic” equations, that nudge the particles toward the observations. We nudge the particles by applying a control $u_k(y, x)$ according to

$$(23) \quad X_{k+1} = F(X_k) + u_k(y_{k+1}, X_k) + \sigma_X W_{k+1}.$$

The technique for determining the nudging term $u_k(y_{k+1}, x)$ is by method of stochastic optimal control by minimizing a quadratic cost:

$$(24) \quad J := \mathbb{E}_{k,x} \frac{1}{2} \left[u_k^T(y, x) Q^{-1} u_k(y, x) + (y - h(X_{k+1}^{(k,x)}))^T R^{-1} (y - h(X_{k+1}^{(k,x)})) \right],$$

where Q , R are the signal and observation noise covariance matrices and by $X^{(k,x)}$ we mean that the process X started at time k at the value x . Again, we assume the linear sensor function with observation available at every timestep, given by (19). The purpose of the present section is to show that control methods can be used as a basic and flexible tool for the construction of the proposal density inherent in particle filtering. In this framework, u can be interpreted as the optimal control for minimizing the cost J . The first term in (24) represents the control energy and if we allow u to become too big, then heuristically all the particles will coincide with the observation. Then the particles will be a sample from a Dirac distribution, whereas the conditional distribution that we try to simulate is absolutely continuous. The second term in (24) represents the distance between HX_{k+1} and the observation that we want minimized. Covariance matrices Q and R in the quadratic terms indicate that dimensions of the signal and observation that have larger noise variance are penalized less by the control. This means that in directions where noise amplitude is large, we allow for more correction by taking Q^{-1} , which puts less penalty on the size of the control in the dimensions with large noise amplitude. Similarly, the terminal cost given by the second term in (24) incurs a penalty for being far away from the actual signal based on observation, but in directions where the quality of the observation is not very good, we allow our particle to be further away from the observation, hence R^{-1} .

Define the value function

$$V^{opt}(k, x) := \inf_{u_k(y, x)} \mathbb{E}_{k,x} \frac{1}{2} \left[u_k^T(y, x) Q^{-1} u_k(y, x) + (y - h(X_{k+1}^{(k,x)}))^T R^{-1} (y - h(X_{k+1}^{(k,x)})) \right].$$

Using (18) and (19), and substituting for $X_{k+1}^{(k,x)}$ in the second expression in the value function, we have

$$\begin{aligned} & \left(y - HX_{k+1}^{(k,x)} \right)^T R^{-1} \left(y - HX_{k+1}^{(k,x)} \right) \\ &= \left(y - H\tilde{F}(x, u_k) \right)^T R^{-1} \left(y - H\tilde{F}(x, u_k) \right) - 2 \left(y - H\tilde{F}(x, u_k) \right)^T R^{-1} (H\sigma_X W_{k+1}) \\ & \quad + (H\sigma_X W_{k+1})^T R^{-1} (H\sigma_X W_{k+1}), \end{aligned}$$

where $\tilde{F}(x, u_k) = F(x) + u_k(y, x)$. Because u_k depends only on y and x , both of which are given, we have $\mathbb{E}_{k,x} [u_k Q^{-1} u_k] = u_k Q^{-1} u_k$, and similarly

$$\mathbb{E}_{k,x} \left(\left(y - H\tilde{F}(x, u_k) \right)^T R^{-1} \left(y - H\tilde{F}(x, u_k) \right) \right) = \left(y - H\tilde{F}(x, u_k) \right)^T R^{-1} \left(y - H\tilde{F}(x, u_k) \right).$$

W_{k+1} is a standard Gaussian random variable independent of $X_k = x$, so

$$\mathbb{E}_{k,x} \left[\left(y - H\tilde{F}(x, u_k) \right)^T R^{-1} (H\sigma_X W_{k+1}) \right] = \left(y - H\tilde{F}(x, u_k) \right)^T R^{-1} H\sigma_X \mathbb{E} [W_{k+1}] = 0,$$

$$\mathbb{E}_{k,x} \left[(H\sigma_X W_{k+1})^T R^{-1} (H\sigma_X W_{k+1}) \right] = \text{tr} \left((H\sigma_X)^T R^{-1} (H\sigma_X) \right)$$

Therefore,

$$V(k, x) = \frac{1}{2} \inf_{u_k} \left\{ u_k^T Q^{-1} u_k + \left(y - H\tilde{F}(x, u_k) \right)^T R^{-1} \left(y - H\tilde{F}(x, u_k) \right) + \text{tr} \left((H\sigma_X)^T R^{-1} (H\sigma_X) \right) \right\}$$

and

$$\begin{aligned} \frac{\partial}{\partial u_k} V(k, x) &= Q^{-1} u_k + H^T R^{-1} H (F(x) + u_k) - H^T R^{-1} y \\ &= (Q^{-1} + H^T R^{-1} H) u_k - H^T R^{-1} (y - HF(x)), \end{aligned}$$

hence the optimal control is

$$u_k^{opt} = (Q^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1} (y - HF(x)),$$

which is similar to the solution from the previous approach, with the difference being the noise term, $\hat{\sigma}_X$ in (21) and σ_X in (23).

This section provided the results related to the control design of the particles (prior to updating the weights) that is needed to nudge the particle solutions toward the observations. This procedure consists of adding, forcing terms to the “prognostic” equations for the construction of the proposal density inherent in particle filtering. However, it is possible that the nudging terms may become too large and destroy the balance between the effects of the noise and the control terms. The stochastic optimal control approach presented in this section is similar to the derivation of the 4D-VAR method that is used in geophysical data assimilation (see, for example, Kalnay [2003]). The 4D-VAR method considers the problem of determining the best initial condition at time t_0 for the forward integration of the model PDEs based on discrete observations collected, up to a finite time t_K , in the future of t_0 . In the 4D-VAR method, the cost function to be minimized with respect to the initial condition $x(t_0)$ is

$$\begin{aligned} J(x(t_0)) &= \frac{1}{2} [x(t_0) - x^b(t_0)]^T B_0^{-1} [x(t_0) - x^b(t_0)] \\ &\quad + \frac{1}{2} [H(x(t_K)) - y(t_K)]^T R^{-1} [H(x(t_K)) - y(t_K)], \end{aligned}$$

where $x^b(t_0)$ was predicted using the model equations from time before t_0 and $x(t_K)$ is obtained by integration of the model PDEs using $x(t_0)$ as initial condition. From this point of view, the stochastic optimal control approach presented here can be viewed as determining the optimal initial condition at every discrete time t_k using the next available observation at t_{k+1} . The optimal control u_k^{opt} is the correction made to the state x_k predicted from t_{k-1} .

The SIS presented in Section 4.2.2 allows the modification of the drift terms as well as the stochastic coefficients in the “prognostic” equations and SIS is an easy approach to implement numerically, therefore we will use SIS in conjunction with HHPF throughout this paper. However, when dealing with sparse data, a proposal density based on stochastic control theory presented in this section is essential, as can be shown in Lingala et al. [2012].

4.2.4. HHPF. In the following we describe how the HHPF uses the SIS algorithm. For now, we would choose the proposal density $q(\cdot | x_{k-1}^i, y_k) = p(\cdot | x_{k-1}^i)$ i.e. the conditional prior which can be obtained from the signal dynamics. A particle can be sampled from this q by propagating the location of the particle at $k-1$ using the signal dynamics. We note that by choosing such a q we are being blind to the observation y_k . We address a better choice in Section 4.2.5.

The HHPF is developed based on the results presented in Section 2. It incorporates the HMM described in Section 4.1 in the particle evolution step of a particle filtering algorithm. Although the problem presented in Section 3 has continuous-time signal, the HHPF is applied on the associated discretized model.

Consider the time interval $[0, T]$, discretized into macro-timesteps of size $\Delta t = \lfloor \frac{T}{N} \rfloor$ and denote the micro-timestep by $\delta t \ll \Delta t$, where δt is chosen small enough compared to ε for numerical stability. In the context of the HMM, the intervals $[k\Delta t, (k+1)\Delta t]$, $k = 1, \dots, N-1$, are the macro-timestep intervals over which evolution of the slow process occurs. The fast process is evolved over micro-timestep intervals $[j\delta t, (j+1)\delta t]$, $j = 1, \dots, n_T + N_m$ within each

macro-timestep interval. The discretized version of the equation (9) governing the coarse-grained dynamics \bar{X}_t , is used for propagation of particles $\{\bar{x}_k^i\}_{i=1}^{N_s}$ over macro-timesteps:

$$(25) \quad X_k = X_{k-1} + \tilde{b}(X_{k-1}) \Delta t + \sigma_x \Delta W_k,$$

where \tilde{b} is defined in (12). Note that (25) is the same as (11) but with additive noise, which is introduced to regularize the system, for the purpose of weight calculation.

Instead of continuous observations, we assume that observations are available at discrete macro-timesteps. Write $Y_k^\varepsilon \stackrel{\text{def}}{=} Y_{t_k}^\varepsilon$. The discrete-time observations are

$$(26) \quad Y_k^\varepsilon = h\left(X_k^\varepsilon, Z_{k, \lfloor \frac{\Delta t}{\delta t} \rfloor}^\varepsilon\right) + \sigma_y V_k.$$

Observations Y_k^ε are used to update the sample by altering particle weights and resampling the system of particles. For notational consistency, we denote the particles representing the averaged \bar{X}_k by \bar{x}_k^i .

The evolution of the particle system via the HHPF is given by the following steps with a graphical illustration of each step shown in Figure 1.

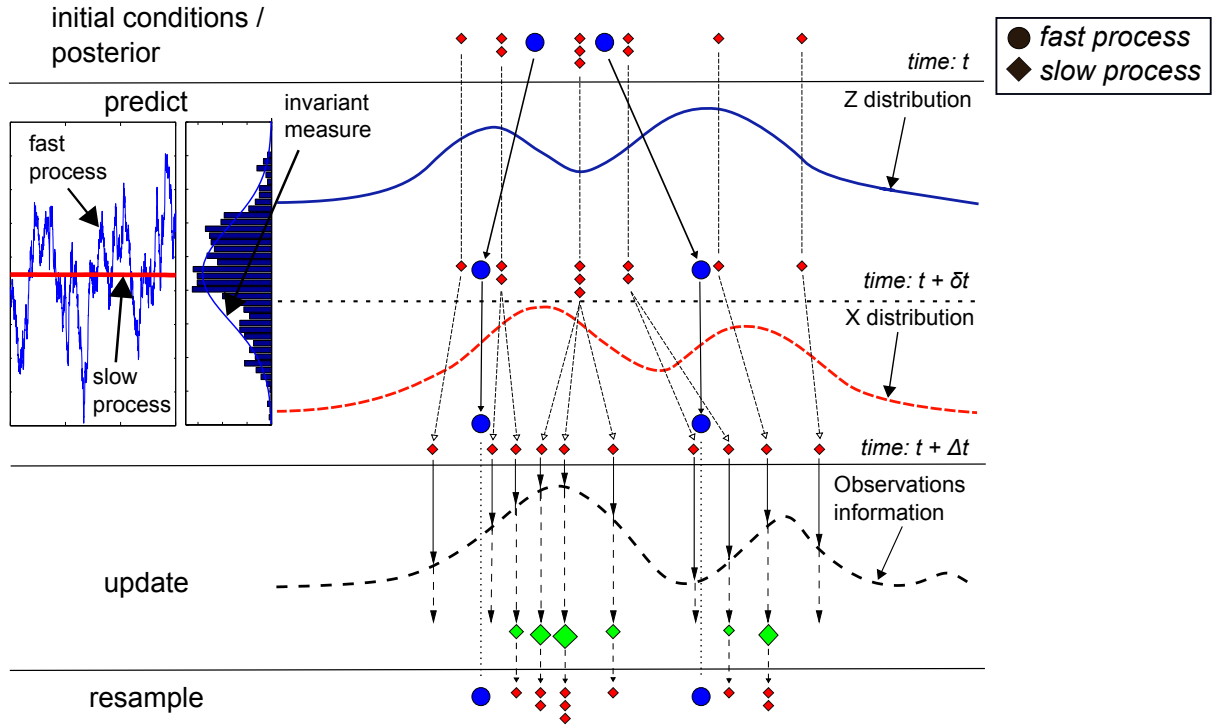


FIGURE 1. HHPF illustrative scheme

- (a) **initial condition:** Samples of N_s particles for the slow process, \bar{X}_k , and $N_s \times M$ particles for the fast process, $Z_{k,j}^\varepsilon$ (subscripts k and j are the macro-step and micro-step indices, respectively, not to be confused with the spatial indices, superscripts k and j , in Section 3), are drawn from the initial distribution of the state variables. The initial distribution of the slow process, π_0^0 , is approximated by a sample of N_s particles $\{\bar{x}_0^i\}_{i=1}^{N_s}$ drawn from π_0^0 , each of mass $\frac{1}{N_s}$, i.e.

$$\pi_0^0 \approx \frac{1}{N_s} \sum_{i=1}^{N_s} \delta(x - \bar{x}_0^i).$$

Similarly, the initial distribution of the fast process is approximated by a collection of M particles $\{z_0^{\varepsilon,i,r}\}_{i,r=1}^{N_s,M}$ which are equally weighted ($z_0^{\varepsilon,i,r} \in \mathbb{R}^n$ is the position

of the r^{th} particle). Note that we are only interested in the coarse-grained dynamics \bar{X}_t of the system. The fast process, assumed to be exponentially mixing, is spatially and temporally averaged through the implementation of the HMM. Thus, under the assumption of ergodicity, we can set $M \ll N_s$ or even $M = 1$ without loss of accuracy in approximating the distribution of \bar{X}_k .

- (b) **prediction and update:** The prediction and update step is where HHPF differs from regular particle filters through the incorporation of dimensional reduction and homogenization techniques.

In a regular particle filter, particles evolve independently according to the discretized signal equations which govern the behaviour over a micro-step δt . Recall that δt is small compared to ε . So, the regular filters simulate the slow process also every δt , even though it does not change much in such a small time, thus wasting the computation resources.

The HHPF utilizes multiscale scheme of the HMM to propagate the sample forward in time. Particles $\{\bar{x}_k^i\}_{i=1}^{N_s}$ are propagated independently over macro-timesteps Δt using the macro-solver (11), where the averaged drift is approximated using (12). Particle propagations in a macro-timestep interval are illustrated in the “predict” segment of Figure 1. Within each macro-timestep interval, the particles representing the *fast* process, $\{z_{k,j}^{\varepsilon,i,r} : k \text{ fixed}\}_{i,r=1}^{N_s,M}$, evolve according to the micro-solver (13) over micro-timestep intervals $[j\delta t, (j+1)\delta t]$, $j = 1, \dots, N_m - 1$, while the *slow* process, $\bar{X}_k = x$ is fixed throughout the macro-timestep interval. Note that in the HMM scheme, $N_m < \frac{\Delta t}{\delta t}$; N_m only needs to be sufficiently large for the fast process to attain an invariant measure within $[k\Delta t, (k+1)\Delta t]$. Also, the particles representing $z_{k,j}^{\varepsilon,i,r}$ that are propagated over the micro-timesteps are in total $N_s \times M$, but M can be set to be 1 by appropriately adjusting the value of N_m , thus the number of fast particles that need to be propagated can be N_s , even with implementation of the HMM scheme. The coefficient \bar{b} in (12) is computed by averaging over the particle locations $\{z_{k,j}^{\varepsilon,i,r} : j = 1, \dots, n_T + N_m\}_{i,r=1}^{N_s,M}$. Averaging over $z_{k,j}^{\varepsilon,i,r}$ is performed spatially over the sample of M particles and temporally over the micro-timesteps interval $[n_T\delta t, (n_T + N_m)\delta t]$. The prediction step is given by (25).

The particles are updated at time-steps when observations are available, as illustrated in the update segment of Figure 1. Only the sample of fast process $\{\bar{x}_k^i\}_{i=1}^{N_s}$ is updated, since, in the multiscale scheme, we are interested only in the dynamics of the homogenized \bar{X}_k . The SIS algorithm is used for updating the sample. In propagating the particles $\{\bar{x}_k^i\}$, as described above, according to the homogenized signal dynamics over Δt , we arrive at new location of the particles $\{\bar{x}_{k+1}^i\}$. These new locations can be thought of as a sample drawn from the proposal density $q(\cdot|x_k)$ which is same as the prior $p(\cdot|x_k)$ of the homogenized dynamics. Particle weights are calculated sequentially according to (15) and because of our assumption of the proposal, we have

$$w_k^i = p(y_k^\varepsilon | \bar{x}_k^i) w_{k-1}^i.$$

We would like to point out another difference of the HHPF compared to a regular SIS filter, that arises due to homogenization.

Considering independent standard Gaussian noise increments for the sensor noise, i.e. $V_k \sim \mathcal{N}(0, \mathbb{I})$, in (26), the likelihood function is Gaussian:

$$(27) \quad p(y_k^\varepsilon | \bar{x}_k) \propto \exp \left\{ -\frac{1}{2} (y_k^\varepsilon - \bar{h}(\bar{x}_k))^T (\sigma_y \sigma_y^T)^{-1} (y_k^\varepsilon - \bar{h}(\bar{x}_k)) \right\}.$$

Instead of the sensor function $h\left(X_k^\varepsilon, Z_{k, \lfloor \frac{\Delta t}{\delta t} \rfloor}^\varepsilon\right)$, the averaged sensor function

$$(28) \quad \bar{h}(x) = \int_{\mathbb{R}^n} h(x, z) \mu_x(dz)$$

is used, since we are dealing with the coarse-grained dynamics. As with the homogenized drift, the averaged sensor function is approximated by \tilde{h} via the HMM. \tilde{h} has the form

$$(29) \quad \tilde{h}(X_k) = \frac{1}{MN_m} \sum_{r=1}^M \sum_{j=n_T}^{n_T+N_m} h(\bar{X}_k, Z_{k,j,r}^\varepsilon).$$

It is worthwhile to note that the actual available observation Y_k^ε is used instead of a fictitious averaged \bar{Y}_k in calculating the weights. Thus the homogenized system (9) is combined with the actual observation Y^ε , from which the name “homogenized hybrid” was derived.

After a few time steps, all the weights may tend to concentrate on a very few particles, which drastically reduces the effective sample size. This issue is addressed in the following step.

- (c) **resampling:** The nature of the sequential importance sampling algorithm is such that the variance of the unnormalized weights increases with each iteration (see, for example, Prop. 3, p. 7 in Doucet [1998], Theorem, p. 285 in Kong et al. [1994]). Due to the gradual increase in weights variance over time, weights will tend to concentrate on a few particles, causing the issue of sample degeneracy. Sample degeneracy decreases the ability of the weighted sample to properly represent the target posterior distribution since only a limited number of particles will have significant weights. Additionally, it incurs the cost of propagating and storing particles with insignificant weights, which effectively do not contribute to representing the target distribution. One method of addressing this issue is to perform occasional resampling when the sample degeneracy level reaches a certain threshold. Resampling does not overcome the issue of weights degeneracy; it only serves to rejuvenate the sample by eliminating particles with insignificant weights and multiplying those with weights that significantly contribute to approximating the posterior.

The measure of sample degeneracy can be determined by the effective sample size N^{eff} (see, for example, Arulampalam et al. [2002]):

$$N_k^{\text{eff}} \stackrel{\text{def}}{=} \frac{N_s}{1 + \text{Var}(\bar{w}_k^*)}, \quad \bar{w}_k^{*i} \stackrel{\text{def}}{=} \frac{p(\bar{x}_k^i | y_{0:k}^\varepsilon)}{q(\bar{x}_k^i | \bar{x}_{k-1}^i, y_k^\varepsilon)},$$

where \bar{w}^{*i} is the “true” weight. The exact values of \bar{w}^{*i} cannot be evaluated since $p(\bar{x}_k^i | y_{0:k}^\varepsilon)$, the actual posterior, is not known, so the effective sample size is approximated numerically using the normalized weights by Arulampalam et al. [2002]

$$\tilde{N}_k^{\text{eff}} = \frac{1}{\sum_{i=1}^M (\bar{w}_k^i)^2}.$$

$\tilde{N}_k^{\text{eff}} \leq N_s$ represents the effective sample size at timestep k , and a resampling procedure is carried out whenever \tilde{N}_k^{eff} falls below a set threshold.

One resampling procedure is the systematic resampling procedure, as described in Arulampalam et al. [2002], Doucet [1998]. The resampling procedure involves sampling with replacement on the current sample, multiplying particles with significant weights and discarding those with insignificant weights. The new sample is reinitialized with uniform weights $\frac{1}{N_s}$ for each particle.

In addition to resampling, the issue of sample degeneracy can also be addressed by choosing a good importance sampling density q . As described so far, the importance

density $q(x_k|x_{k-1})$ is based on the propagation of the sample from the previous estimation step. i.e. the observation Y_k was not involved in the importance density at time k . In Section 4.2.5 below, we propose a method of specifying an importance density $q(x_k|x_{k-1}, y_k)$ at time k that makes use of the observation at time k .

Algorithm 1 shows a pseudo code for the HHPF. Overall, the advantages of Algorithm 1 are:

- (a) The number of fast samples evaluations is greatly reduced (if the number of fast sample replicas is set as $M = 1$ by choosing N_m appropriately), since $N_m < \lfloor \Delta t / \delta t \rfloor$.
- (b) The total number of timesteps is decreased due to the relatively large macro-timesteps.
- (c) The number of function evaluations are also decreased accordingly.

Even though there are additional function evaluations (as in equation (12)) in incorporating the HMM, these are negligible compared to the original weight calculations of the regular branching particle method, which have been reduced in the HHPF through (29).

Although the HHPF has been adapted for multiscale computations, another issue arises in its application on complex problems with inherent chaotic nature such as that described in Section 3. As shown in Section 5, we are able to perform state estimation for a chaotic system using the HHPF as described so far, the procedure can be made more efficient by importance sampling. However, as mentioned earlier, a key aspect of the importance sampling algorithm is the choice of the sampling density q . In the following section, we present a method of constructing a good sampling density by introducing a forcing term in the particle evolution of (25).

```

Draw samples from initial distribution:  $\{\bar{x}_{k=1}^i\}_{i=1}^{N_s}, \{z_{k=1,j=1}^{\varepsilon,i,r}\}_{i,r=1}^{N_s,M}$ 
for k = 1:number of macro-timesteps ( $K$ ) do
  for r = 1:number of replicas ( $M$ ) do
    for j = 1:number of micro-timesteps ( $N_m$ ) do
      Solve micro-solver (13):  $z_{k,j+1}^{\varepsilon,i,r}$ 
    end for
  end for
  Perform averaging (12) and (29):  $\{\tilde{f}(\bar{x}_k^i), \tilde{h}(\bar{x}_k^i)\}_{i=1}^{N_s}$ 
  Solve macro-solver (11):  $\{\bar{x}_{k+1}^i\}_{i=1}^{N_s}$ 
  Compute weights:  $\{\bar{w}_{k+1}^i\}_{i=1}^{N_s}$ 
  Compute effective sample size:  $\tilde{N}_{\text{eff}}^{k+1}$ 
  Resample using choice of resampling algorithm if  $\tilde{N}_{\text{eff}}^{k+1} < N_{\text{thres}}$ 
  Reinitialize:  $z_{k+1,j=1}^{\varepsilon,i,r} = z_{k,N_m}^{\varepsilon,i,r}$ 
end for

```

Algorithm 1: HHPF (Park et al. [2011])

4.2.5. *HHPF using the optimal proposal.* In the event that the averaged sensor function (28) is expressible as $\bar{h}(x) = Hx + C_k$, where C_k is a vector which can change with time k , then one can use the optimal proposal which keeps the variance of weights to minimum, as explained in 4.2.2. Note that \bar{h} is of this form if the observation function h is linear and only depends on the slow components. Only the following changes need to be made to the algorithm:

- Instead of propagating the particles \bar{x}_{k+1}^i by using (25), we propagate according to

$$(30) \quad X_k = X_{k-1} + \tilde{b}(X_{k-1}) \Delta t + \alpha(X_k, y_k - C_k) + \hat{\sigma}_x \Delta W_k,$$

where $\hat{\sigma}_x$ is such that $\hat{Q} \stackrel{\text{def}}{=} (\hat{\sigma}_x \hat{\sigma}_x^T) = ((\sigma_x \sigma_x^T)^{-1} + H^T (\sigma_y \sigma_y^T)^{-1} H)^{-1}$, and $\alpha(X_k, y_k) = \hat{Q} H^T R^{-1} (y_k - H f(X_{k-1}))$, where $f(x) = (x + \tilde{b}(x) \Delta t)$.

- The weights are updated according to

$$w_k^i \propto w_{k-1}^i \exp \left\{ -\frac{1}{2} (y_k - C_k - H f(x_{k-1}^i))^T \hat{R}^{-1} (y_k - C_k - H f(x_{k-1}^i)) \right\},$$

$$(31) \quad \hat{R}^{-1} = R^{-1} \left(\mathbf{1} - H \hat{Q} H^T R^{-1} \right).$$

5. APPLICATION

Based on the results of homogenization and optimal importance sampling, we have developed a new lower-dimensional particle filter, the HHPF, for state estimation in nonlinear multi-scale systems. In this section, we illustrate the HHPF's potential for state estimation in a high-dimensional complex problem by applying the HHPF algorithm to the Lorenz '96 (Lorenz [1995]) atmospheric model (see Section 3) to estimate the *slow* variables. The HHPF algorithm is implemented in discrete time, using SIS, as presented in Section 4.2. The model parameters for application of the HHPF are as follows: $K = 36$, $J = 10$, $F_x = 10$, $h_x = -0.8$, $h_z = 1$, $\varepsilon = 1/128$.

The model was first simulated for 40960 timesteps of size 2^{-11} starting from arbitrary initial conditions to represent the “true” signal X^ε that was to be estimated. The timestep value of 2^{-11} was picked such that it is small enough compared to the separation parameter ε to ensure numerical stability. The Lorenz '96 model given by (5), (7) was integrated using a split timestepping scheme: fourth order Runge-Kutta scheme for the deterministic drift and the Euler-Maruyama scheme for the stochastic parts. These schemes were selected for simplicity of implementation in the numerical experiments, but of course, coarser timesteps may be used by implementing higher order integration schemes. Observations generated from the true states were of the form assumed in Section 4.2.5: $Y_t^\varepsilon = H X_t^\varepsilon + B_t$, i.e. depends linearly on X^ε , perturbed by a standard Gaussian noise. Observations were generated every 128 timesteps, i.e. at every timestep of size 2^{-4} . This is the size of the macro-timestep Δt that we select for the HMM integration scheme. This means that we are considering the case where observations are available sequentially (at every timestep) at the timescale that we had chosen for the numerical integration of the slow process. In Lorenz [1995], the timestep chosen for numerical integration of the multiscale system with $\varepsilon = 0.01$ was 0.05, which corresponds to 36 minutes in real time. We follow Fatkullin and Vanden-Eijnden [2004] in selecting $\Delta t = 2^{-4}$ and $\delta t = 2^{-11}$ for the HMM scheme for $\varepsilon = 2^{-7}$. Based on Lorenz [1995]'s scale, these timesteps approximately correspond to 45 minutes and 35 seconds, respectively, in real time and the duration of the model simulation and data assimilation experiments correspond to 10 days (14400 minutes). On the real time scale, the assumption of sequentially available observation data corresponds to observation data collection every 45 minutes, which is not an unrealistic assumption.

We consider two cases for the linear observation matrix H . The first is $H = I_{K \times K}$, where I is the identity matrix, i.e. all slow states are observed (non-sparse observations). The second is $H_{ij} = 1$ if $i = j$ and i odd, and otherwise $H_{ij} = 0$, i.e. only the odd-indexed slow dimensions are observed (sparse observations). Observation noise covariance matrix is assumed to be the identity matrix, i.e. observation noise in each dimension is independent of the rest. We chose the slow and fast signal noise covariance matrices to be 1 on the diagonals and 0.5 on the sub- and super-diagonals as in van Leeuwen [2010], with the fast noise scaled by $\varepsilon^{-1/2}$. Initial conditions for the true signal are arbitrarily chosen from mean zero normal distributions with variances 3 and 5 for the slow and fast processes, respectively. The filtering objective is to estimate the *slow* states using sequentially available observations. We discuss the numerical experiments in the following. The numerical experiments were performed using MATLAB v. 7.11.0.584 (R2010b), without explicitly employing parallel processing capabilities, on an Intel Xeon DP Hexacore X5675 3.07 GHz processor with 12×4 Gb RAM.

5.0.6. Optimized HHPF with linear observations. In the first set of numerical experiments, we consider the case of non-sparse observations and consider two variants of the HHPF to study the effects of the optimal particle propagation for linear observation functions discussed in Section 4.2.5. The SIS algorithm is used in both variants of the HHPF. In one variant, the proposal density used for the importance sampling procedure is generated using particles propagated using the optimal drift and diffusion as given in (20), with weights updated accordingly. In the other variant, proposal density is the prior density generated by propagating particles directly

according to (25). We call the first one the *optimized HHPF* and the second one the *direct HHPF*. For the HMM scheme, the HMM window was set at $m_N = 64$ microtimesteps, and number of microtimesteps skipped to ignore transient effects is $m_T = 32$. The number of replicas is set at $m_r = 1$.

Numerical experiments were performed using both HHPFs with varying sample sizes starting from $N_s = 2$ to $N_s = 400$. Both filters achieved better accuracy with increasing N_s , as they should, and Figures 2 and 3 each show the comparison of one dimension of the true state with its estimates using the HHPFs and the corresponding estimation errors. In both upper figures, blue curve represents the true state, broken red curve represents the estimate using the optimized HHPF, and green curve represents that of the direct HHPF. In the lower figures, blue curve represents the observation error, broken red curve and blue curves represent the estimation error of the optimized and direct HHPFs, respectively. The error shown is the absolute error over all 36 slow dimensions, i.e. $\text{error}_t = \sqrt{\sum_{k=1}^{36} (\hat{X}_t^k - X_t^{k,\varepsilon})^2}$. Figure 2 shows the HHPFs using *just 2 particles* and the optimized HHPF was seen to perform fairly well in estimating the true state, and the error plot showed that the estimate was only slightly worse than the observation error. This is not surprising, since all the slow states were observed and the particles were driven towards those states as indicated by the observations. Estimation error was contributed to in part by the stochasticity in the system and the observation noise, as well as the error due to approximation by numerical homogenization. The direct HHPF however fails to estimate the truth with $N_s = 2$. The estimate initially converged to the truth but diverges at around $t = 2.5$ and fails to recapture it after that. Both the optimized and direct HHPFs took just 5 seconds to run over the entire interval of 320 macrotimesteps, which is a good performance for the optimized HHPF. But considering that the estimates were at best only as good as the noisy observations, the most computationally efficient choice would be to just use observations directly if observations of all states are available.

By increasing the number of particles used, we see in the error plot of Figure 3 that the optimized HHPF was able to provide a better estimate of the truth than what was known directly from the observations. The run time for the experiment presented in the figure was 135 seconds, which is the typical time recorded for the optimized HHPF with $N_s = 100$. The estimate from the direct HHPF improved but was at best as good as the observation with $N_s = 100$. Increasing N_s up to 400 improved both filters' estimates but not significantly for the optimized HHPF. The direct HHPF was seen to be able to match the optimized HHPF with $N_s = 100$ by using 600 particles but the run time required was 639 seconds. The second and third columns of Table 1 show the run times for different sample sizes for a typical experiment using the HHPFs. For each fixed N_s , run times for the optimized and direct HHPFs were observed to be within the same range, which was expected, since the only additional function evaluations in the optimized version is the calculation of the "nudging" correction that is just a linear combination of observation vector with particle locations. Occasional drastic variation in run times between the optimized and direct HHPFs, for example for $N_s = 100$ and $N_s = 400$, can be explained based on the number of times the resampling procedure were required to be performed in each filter. In each case, resampling may have been required more frequently due to the initial sample being significantly far from the truth, hence more resampling had to be performed in the beginning of the run, or at some point in time, a large number of particles were forced significantly far from the true state by the stochastic forcing. The second scenario is less likely to happen since the signal noise amplitude is small in comparison with that of the states. Resampling, however, will not drastically increase for the optimized HHPF because particles are simultaneously driven towards the truth based on the observations through the optimal propagation procedure.

The same numerical experiments were also performed for the case of sparse observations (i.e. observing only the dimensions with odd index), to study the performance of the optimized HHPF in estimating hidden states. The same trend as for the case of non-sparse observations was observed in comparing the estimates using the optimized and direct HHPFs. For fixed N_s ,

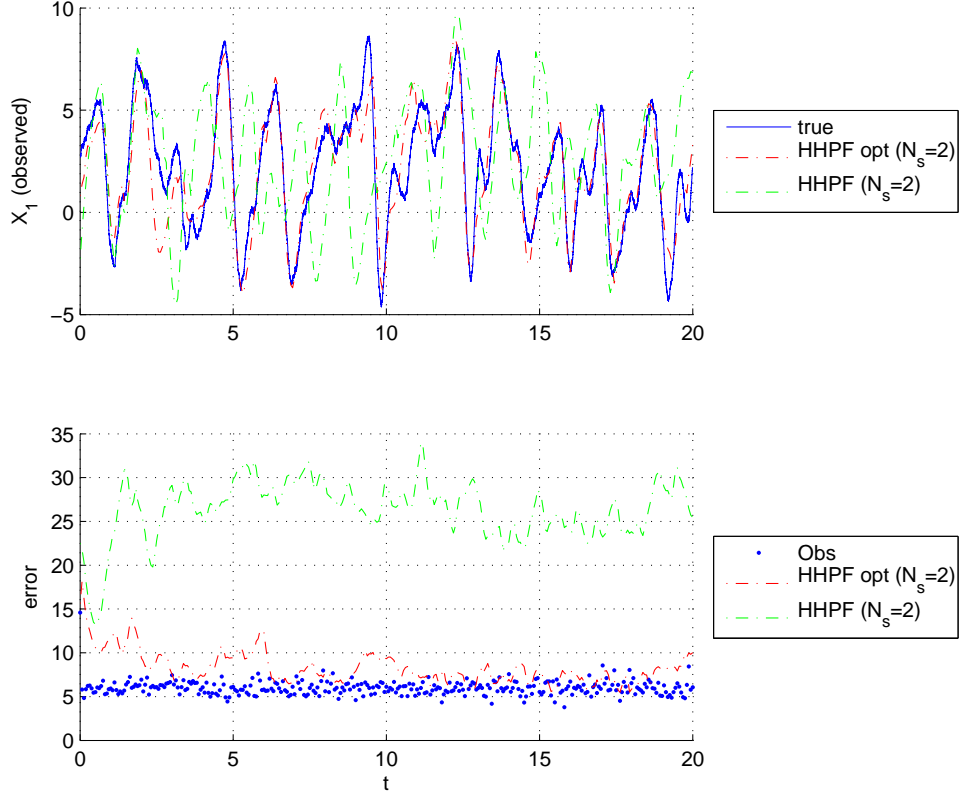


FIGURE 2. Non-sparse observations, optimized and direct HHPFs, $N_s = 2$. The optimized HHPF estimated the truth fairly well but was at best as good as observations. Increasing N_s led to estimates that were better than observation data.

the optimized HHPF achieved better accuracy than the direct version, although both estimates displayed higher estimation errors compared to the non-sparse observation case at the same N_s due to the presence of hidden states. For this case, the optimized HHPF estimate was able to match the observation data of the observed states using 20 particles, and estimate the unobserved states as well. Figure 4 shows the comparisons of the estimates from the HHPFs, using 100 particles for the optimized and 400 particles for the direct, with the truth. The upper plot shows the estimates of an observed state and the lower shows those of an unobserved state. The optimized HHPF (broken red curve) provided good estimates of all the states, including the unobserved ones and the corresponding error plot in Figure 5 shows that the estimation was as good as the observation (of course, observations only observed half the states and the error shown only compares error in the observed dimensions; the HHPF estimates included the unobserved dimensions, shown to capture the truth in the lower plot in Figure 4). The direct HHPF, was also able to capture the shape of the true fluctuations in the observed and unobserved dimensions, but in the observed dimensions, the estimates were poorer than the observations, even with 400 particles. Increasing sample size further up to $N_s = 800$ did not lead to significant improvement. The run times and the trend of increase in run time with sample size for the sparse observations experiments were the same as that of the non-sparse observations experiment shown in Table 1.

The homogenized filters comparison experiments showed that the optimized HHPF displayed significant estimation performance over the direct version for a fixed sample size. This indirectly led to improvement in computation time in comparison with the direct HHPF, in the sense that for the same or even better level of estimation error, the optimized HHPF could be implemented using smaller sample size than the direct version, hence reducing computational costs. In the

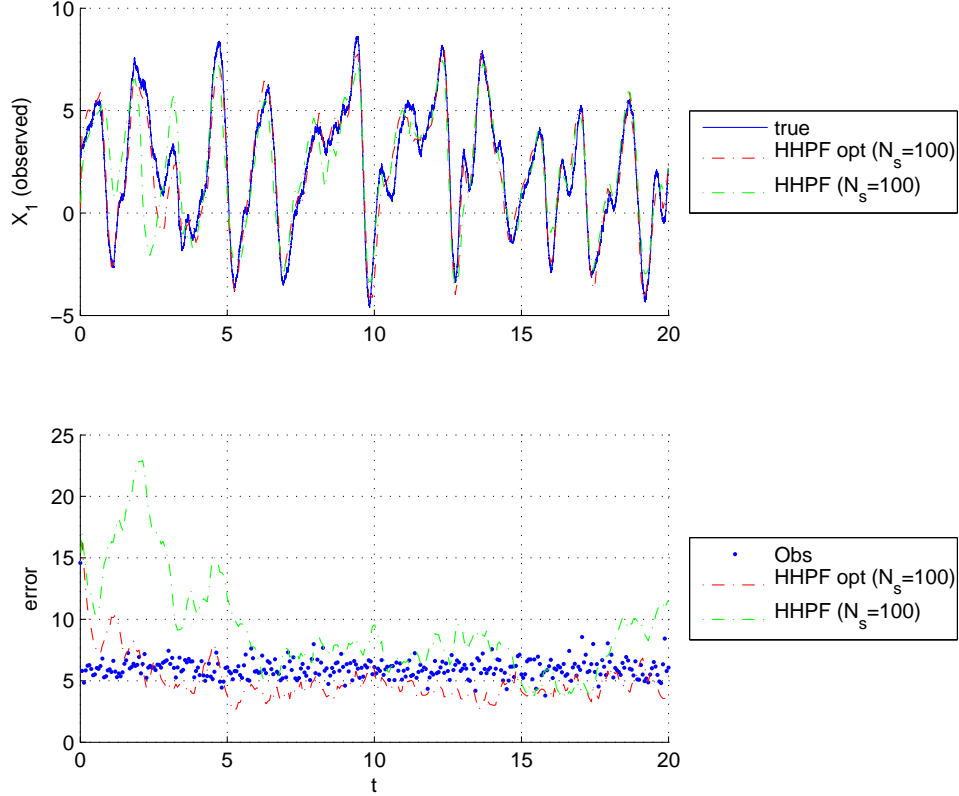


FIGURE 3. Non-sparse observations, optimized and direct HHPFs, $N_s = 100$. The optimized HHPF estimate was better than observation data but the direct HHPF was still not as good. The direct HHPF was able to match the optimized one with $N_s = 600$ at the cost of longer computation time.

discussion here and in Table 1, we have only considered the comparisons of the HHPFs with N_s up to 400, beyond which the run time advantage of the HHPF over other unhomogenized nonlinear filters, for the same level of accuracy, is lost. In the next set of experiments, we compare the optimized HHPF with two other nonlinear filters.

5.0.7. Comparison of the optimized HHPF with other nonlinear filtering schemes. In the second set of numerical experiments, we compared the optimized HHPF with two other nonlinear filtering schemes: the ensemble Kalman filter (enKF) and a particle filter without homogenization. The aim was to compare the run time reduction in the HHPF, due to the implementation of a homogenization scheme, relative to unhomogenized nonlinear filters and assess the trade-off in estimation accuracy due to homogenization. We do not expect the optimized HHPF to outperform an unhomogenized filter in terms of estimation accuracy, but the optimized HHPF should possess comparable estimation capabilities to unhomogenized filters, with the advantage of the HHPF being that it requires shorter run times.

Since observations were generated at macrotimesteps of size 2^{-4} , we have observations being sparse in time as well for the unhomogenized filters, which use the same timestep size of 2^{-11} as the model simulation. The enKF Evensen and van Leeuwen [1996] was implemented directly without modifications, by propagating the ensemble forward in time according to the model dynamics and performing information updates at timesteps when observations are available. For the particle filter, we implement a modified SIS particle filter algorithm developed by van Leeuwen [2010] that is designed to accommodate for observations that are sparse in time. This particle filter is similar to the optimized HHPF presented here, in the sense that it uses the presently available observation to construct a better proposal density at a present time by

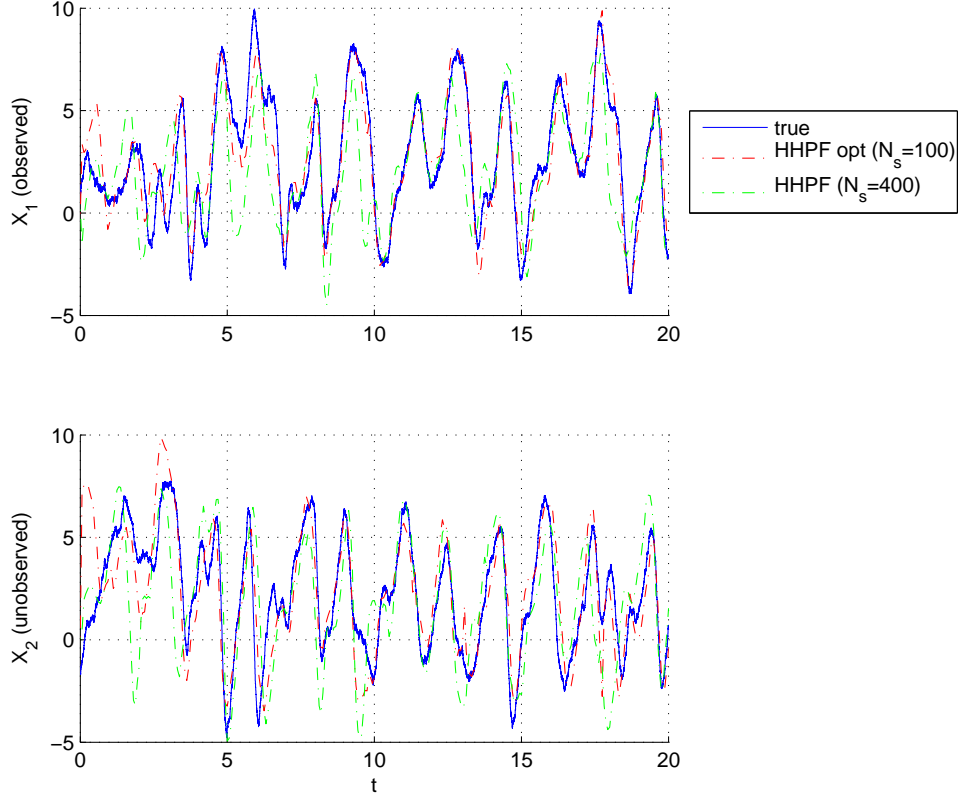


FIGURE 4. Sparse observations, optimized ($N_s = 100$) and direct ($N_s = 400$) HHPFs. The upper plot shows the estimates of an observed state, the lower plot shows that of an unobserved state. The unobserved state was estimated well by the optimized HHPF with $N_s = 100$. Even with $N_s = 400$, The direct HHPF captures the fluctuations in the truth but did not follow the trajectory well.

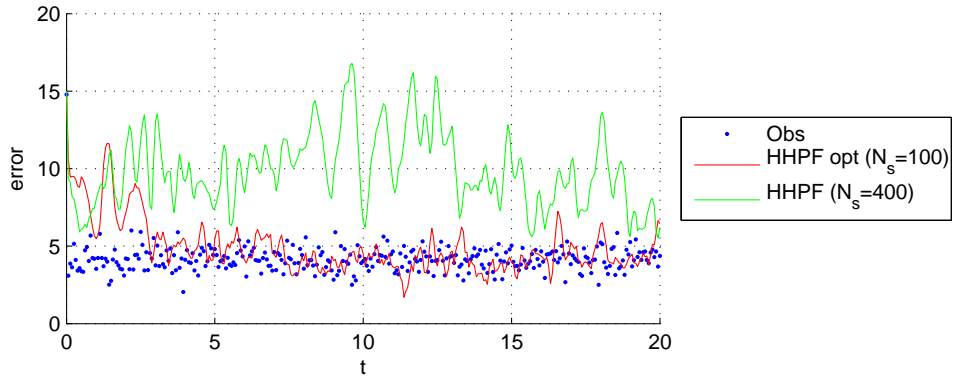


FIGURE 5. Sparse observations, estimation error of the observed states for the optimized ($N_s = 100$) and direct ($N_s = 400$) HHPFs compared with observation error. The optimized HHPF estimates were as good as observations (but the optimized HHPF also provided estimates of the unobserved states so more information is gained by using the filter instead of just observation).

driving particles in between observation timesteps using a time exponential function that is proportional to the model noise covariance and the distance of the intermediate particle locations from the observed state. For details and better insight to this particle filter, see van Leeuwen

[2010], and from here on, we will denote this particle filter as just PF, but it is implied that it is the particle filter adapted for sparse-in-time observations.

Similar numerical experiments as for the comparison of the optimized and direct HHPFs were performed for the EnKF and the PF and the estimation results were compared with those of the HHPFs. We will first discuss the case of non-sparse (spatially) observations, i.e. the case $H = I_{K \times K}$. The PF was able to provide good estimates of the truth with a sample size of just 2. Estimation error decreased as N_s was increased to 20 and was not seen to further decrease significantly as N_s was increased from 20. Figure 6 shows the estimates of one dimension of the true state using the EnKF, the optimized HHPF, and the PF with $N_s = 20$, and their corresponding estimation errors. The blue curve is for the truth, the black for the EnKF, red for the optimized HHPF, and green for the PF. The optimized HHPF displayed the highest estimation error, its estimate being as good as the observation. The EnKF and PF estimates are better than the observation, with estimation errors of equal magnitude. However, when considering the run times, the optimized HHPF took 30 seconds while the EnKF and PF took 540 and 1757 seconds, respectively. Even with $N_s = 2$, the PF took 1727 seconds due to the timestep size and the functional evaluations required for particle weight calculations, as well as the resampling procedures. Additionally, based on the error plot in Figure 6, the estimation error of the optimized HHPF was not much worse than those of the EnKF and PF, and the estimate trajectory followed the truth very well apart from slight over- and under-shoots at local maxima and minima. This indicates that the optimized HHPF required less run time compared to the unhomogenized filters at a comparable level of estimation error. Increasing sample size showed that the optimized HHPF was almost as good as the PF at $N_s = 100$, as shown in Figure 7, with the EnKF being slightly better than both particle filters. The key point is that using 100 particles, the optimized HHPF took 134 seconds while the EnKF and PF took 2169 and 920 seconds, respectively. Considering that the levels of estimation errors are almost equal, the optimized HHPF provided significant advantage in terms of computation time. Further increasing the sample size to 400 enabled the optimized HHPF to match the PF. Beyond $N_s = 400$ however, the optimized HHPF lost its computation time advantage, for the EnKF could be implemented using $N_s = 50$ or 100 with about the same level of estimation accuracy and computation time.

Similar numerical experiments were again performed for the case of sparse observations (i.e. observing dimensions with odd index) and Figures 8 and 9 show the comparisons of the estimates of the observed and unobserved states from the filters using 20 and 100 particles, respectively. With 20 particles, the EnKF and PF provided slightly better estimates of the observed states than the observation, but the optimized HHPF performed rather poorly in comparison. However, at $N_s = 100$, the optimized HHPF's performance improved significantly, performing as well as the PF. Figure 10 shows the estimation error over the observed dimensions and, to compare the filter estimate over all slow dimensions, Figure 11 shows the estimation error over all slow dimensions for each filter. The PF estimate converged to the truth faster but the optimized HHPF estimate eventually became as good as the PF's, with the EnKF's being the best. The key point is again that the optimized HHPF took 134 seconds while the EnKF and PF took 920 and 2169 seconds respectively. Even if we considered using the EnKF with $N_s = 20$ (540 seconds) and the PF with $N_s = 2$ (1727 seconds), both of which provided relatively low-error estimates, the optimized HHPF with $N_s = 100$ is still faster. So, even in the sparse observations case, the optimized HHPF could be implemented in shorter time with estimation error comparable or equal to the EnKF and the PF.

We do not claim that the HHPF is better than the EnKF or the PF; as shown, the unhomogenized filters provided lower estimation error than the homogenized filter at low fixed N_s . As mentioned in the discussion of the previous set of experiments, the accuracy of the optimized HHPF could be increased by increasing N_s , but beyond $N_s = 400$, it loses computation time advantage over the EnKF. However, in terms of computation time and cost of storage, the optimized HHPF displayed advantage over the unhomogenized filters, with comparable level of accuracy. The EnKF would have been the better choice of filter if the lowest possible estimation

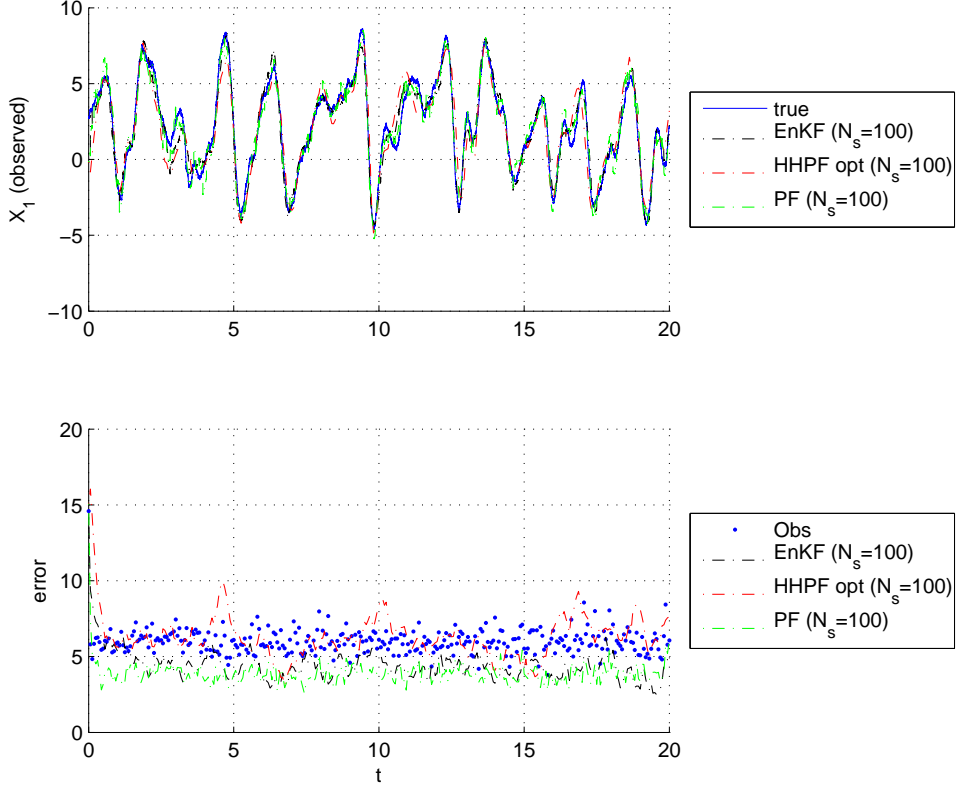


FIGURE 6. Non-sparse observations, EnKF, PF and optimized HHPF comparison at fixed $N_s = 20$. The estimate of the optimized HHPF is comparable to those of the EnKF and the PF, but is obtained in a shorter run time.

error was required over computational time. Additionally, the PF was designed to accommodate temporally sparse observations, as it has been shown to do in van Leeuwen [2010] and here. This capability still needs to be incorporated in the HHPF as presented here, because in most real time applications, the time interval for availability of observation data can be greater than the 45 minute-interval assumed here.

In comparing the computational times of the numerical experiments using the EnKF, the PF and the optimized HHPF so far, we have not taken into account the cost of computing the homogenized observation function \bar{H} . In addition to H being a constant matrix, we also assumed that the sensor observed only the slow states, hence the observation function was independent of the fast process. So, the homogenized observation function is the same as the unhomogenized one. However, we do not expect the cost of evaluating the homogenized observation function to drastically affect the computational time advantage of the optimized HHPF over other unhomogenized nonlinear filters.

6. CONCLUSION AND FUTURE DIRECTIONS

6.0.8. Future directions. In chaotic systems, such as the one studied in this paper, the transients become irrelevant from the dynamical systems point of view and the motion of the solution settles typically near a subset of the state space, called an attractor. However, in the data assimilation problem that is of interest in this paper, we are interested in the transients and, in particular, in directions that are stretched by the transient dynamics. This sensitivity to initial conditions are characterized by the finite time Lyapunov exponents, which are determined by the behavior of two neighboring orbits or the two point motion of the nonlinear systems.

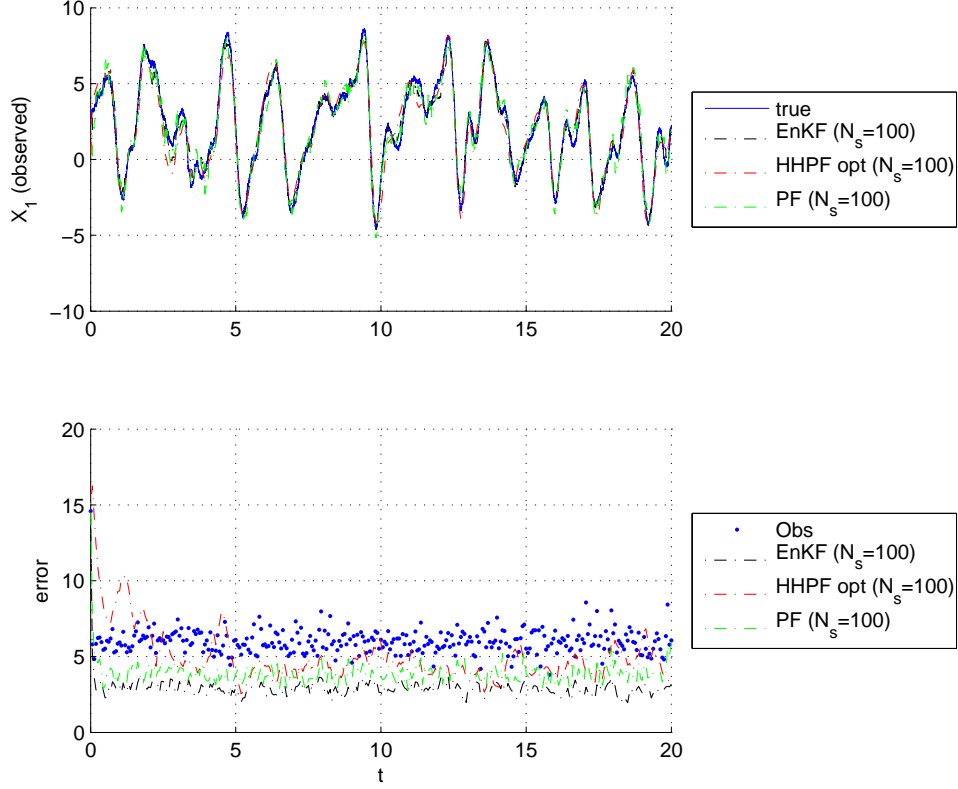


FIGURE 7. Non-sparse observations, estimation error comparison the EnKF, PF and optimized HHPF comparison at fixed $N_s = 100$. The optimized HHPF could match the PF, with the EnKF being slightly better, but the optimized HHPF required the least computation time.

N_s	Opt. HHPF	Direct HHPF	PF	EnKF
2	5	5	1727	N/A
10	16	15	1716	N/A
20	30	20	1757	540
50	67	46	1936	651
100	134	86	2169	920
200	177	176	3002	1415
400	401	539	3591	3152

TABLE 1. Typical computation times (in sec.) for different sample sizes for the different nonlinear filters, in the case of non-sparse observations. We see that for a fixed N_s , the HHPF required less computational time. Circled are the computation times corresponding to the sample sizes that led to the same levels of estimation accuracy for the three different filters compared in Section 5.0.7. For the same level of accuracy, the optimized HHPF required a larger sample size but still performed in less time. Times for the case of sparse observations are of the same magnitudes and display the same trend.

We are mostly interested in filtering *deterministic* chaotic systems, and the particle filtering methods developed above won't work without the addition of noise, because knowing the initial conditions X_0 , the distribution of X_t is a Dirac measure. Therefore we add Gaussian noise

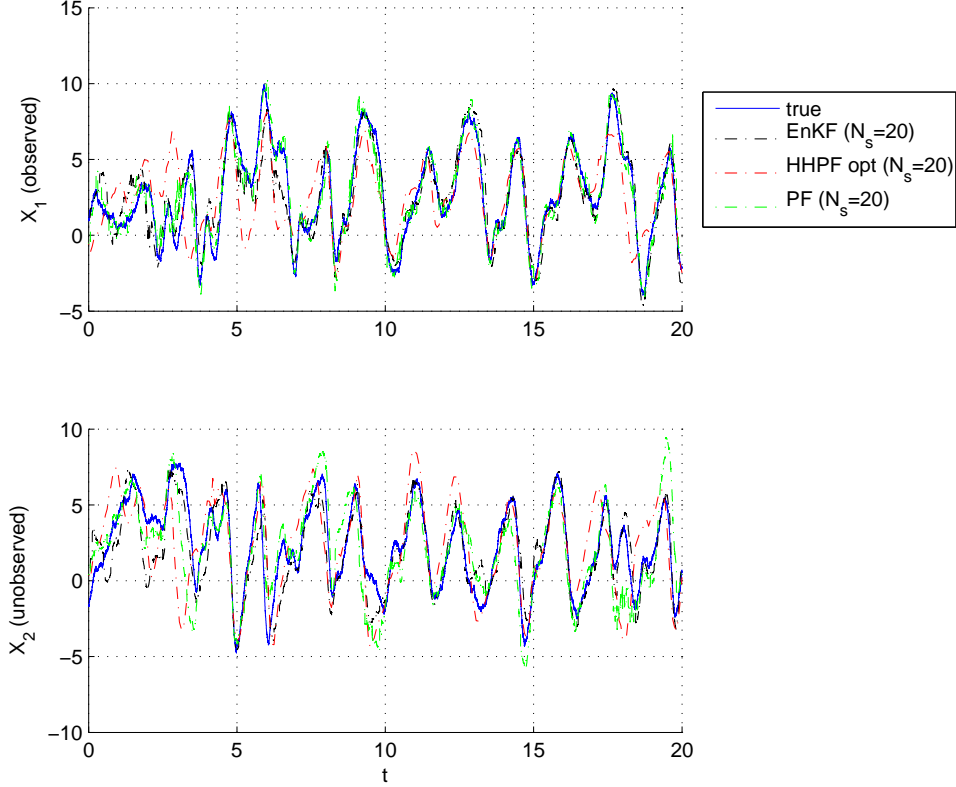


FIGURE 8. Sparse observations, EnKF, PF and optimized HHPF comparison at fixed $N_s = 20$. The optimized HHPF did not perform as well as the EnKF or PF.

artificially. Hence, the finite time Lyapunov exponents may be handy in deciding how we choose the magnitude of the noise.

The second question then arises as to the property of the sensor function, in the linear case, the span of the observation matrix. As pointed out by Lorenz Lorenz and Emanuel [1998] and Palmer et al. [1998], who have coined the word “adaptive” or “targeted” observations, the sensors should be deployed at any given time, if the data that they gather are to be most effective in improving the analysis and forecasts.

The results presented in this paper assume that the observations are available every 45 minutes. The extension of this work that deals with every combination of the spatial and temporal sparsities, performing intermittent in time sparse data assimilation that mimics a global weather model is presented in Lingala et al. [2012]. The particle method presented in Lingala et al. [2012] consists of control terms in the “prognostic” equations, that nudge the particles toward the observations, specially in the sparsest situation of $\frac{K}{4}$ observations in every 48 hours, and shows that control methods can be used as a basic and flexible tool for the construction of the proposal density inherent in particle filtering.

Acknowledgement. Nishanth Lingala, N. Sri Namachchivaya, and Hoong C. Yeong are supported by the National Science Foundation under grant number EFRI 10-24772 and by AFOSR under grant number FA9550-08-1-0206. Nicolas Perkowski is supported by a Ph.D. scholarship of the Berlin Mathematical School. Part of this research was carried out while Nicolas Perkowski was visiting the Department of Aerospace Engineering of University of Illinois at Urbana-Champaign. He is grateful for the hospitality at UIUC. The visit of Nicolas Perkowski was funded by NSF grant number EFRI 10-24772 and by the Berlin Mathematical School. Any

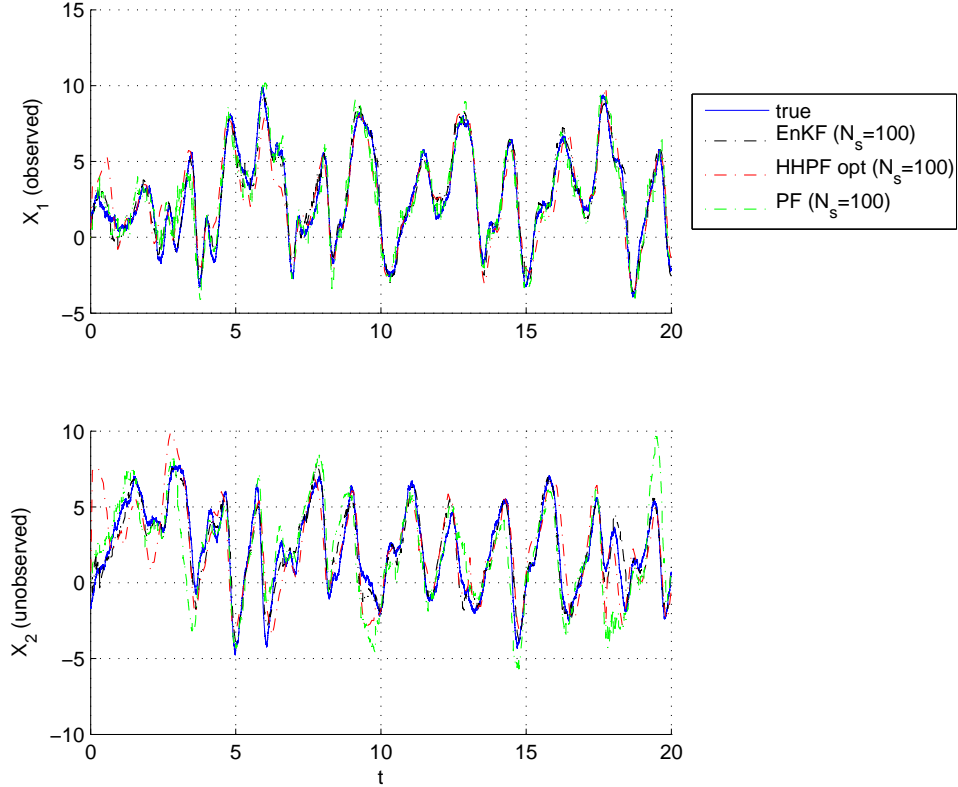


FIGURE 9. Sparse observations, EnKF, PF and optimized HHPF comparison at fixed $N_s = 100$. The optimized HHPF performed as well as the PF, but in shorter time than both the EnKF and PF.

opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

- M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, February 2002.
- A. Bain and D. Crisan. *Fundamentals of stochastic filtering*. Springer Verlag, 2009.
- F. Daum and J. Huang. Curse of dimensionality and particle filters. In *Aerospace Conference, 2003. Proceedings. 2003 IEEE*, volume 4, pages 1979–1993. IEEE, 2003.
- A. Doucet. On sequential simulation-based methods for Bayesian filtering. Technical report, Cambridge University, 1998.
- W. E, D. Liu, and E. Vanden-Eijnden. Analysis of multiscale methods for stochastic differential equations. *Commun. Pur. Appl. Math.*, 58:1544–1585, January 2005.
- G. Evensen and P. J. van Leeuwen. Assimilation of Geosat Altimeter Data for the Agulhas Current Using the Ensemble Kalman Filter with a Quasigeostrophic Model. *Monthly Weather Review*, 124:85–96, January 1996.
- I. Fatkullin and E. Vanden-Eijnden. A computational strategy for multiscale systems., *Journal of Computational Physics*, 200(2):605–638, November 2004.
- N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian Bayesian state estimation. In *IEEE Proceedings-F*, volume 120, pages 107–113, 1993.
- S. Herrera, D. Paz, J. Fernandez, and M. A. Rodriguez. The role of large-scale spatial patterns in the chaotic amplification of perturbations in a lorenz’96 model. *Tellus*, 63(5):978–990, October 2011.
- P. Imkeller, N. S. Namachchivaya, N. Perkowski, and H. Yeong. A homogenization approach to multiscale filtering. Submitted to the Proceedings of the IUTAM Symposium on 50 Years of Chaos: Applied and Theoretical, 2011.
- P. Imkeller, N. S. Namachchivaya, N. Perkowski, and H. Yeong. Dimensional reduction in nonlinear filtering: A homogenization approach. Submitted, 2012.
- E. Kalnay. *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 2003.



FIGURE 10. Sparse observations, comparison of estimation errors of the EnKF, PF and optimized HHPF with observation error.

- P. E. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*. Springer Verlag, Berlin, Heidelberg, 1992.
- A. Kong, J. S. Liu, and W. H. Wong. Sequential Imputations and Bayesian Missing Data Problems. *Journal of the American Statistical Association*, 89(425):278–288, March 1994.
- N. Lingala, N. S. Namachchivaya, N. Perkowski, and H. Yeong. Particle filtering based on optimal nudging of the particle system. In preparation for the Proceedings of the IUTAM Symposium on Multiscale Problems in Stochastic Mechanics, 2012.
- E. Lorenz and K. Emanuel. Optimal sites for supplementary weather observations: Simulation with a small model. *Journal of Atmospheric Sciences*, 55:399–414, February 1998.
- E. N. Lorenz. Predictability: A problem partly solved, 1995.
- A. J. Majda, I. Timofeyev, and E. Vanden-Eijnden. A Mathematical Framework for Stochastic Climate Models. *Communications on Pure and Applied Mathematics*, LIV:891–974, 2001.
- A. J. Majda, I. Timofeyev, and E. Vanden-Eijnden. Systematic Strategies for Stochastic Mode Reduction in Climate. *Journal of Atmospheric Sciences*, 60:1705–1722, July 2003.
- T. N. Palmer, R. Gelaro, J. Barkmeijer, and R. Buizza. Singular vectors, metrics, and adaptive observations. *Journal of Atmospheric Sciences*, 55:633–653, February 1998.
- E. Pardoux. Stochastic partial differential equations and filtering of diffusion processes. *Stochastics*, 3:127–167, 1979.
- E. Pardoux and A. Y. Veretennikov. On Poisson equation and diffusion approximation 2. *Ann. Probab.*, 31(3): 1166–1192, July 2003.
- J. Park, R. B. Sowers, and N. S. Namachchivaya. Dimensional reduction in nonlinear filtering. *Nonlinearity*, 23(2):305–324, Feb. 2010.
- J. Park, N. S. Namachchivaya, and H. C. Yeong. Particle Filters in a Multiscale Environment: Homogenized Hybrid Particle Filter. *J. Appl. Mech.*, 78(6):1–10, November 2011.
- C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson. Obstacles to high-dimensional particle filtering. *Monthly Weather Review*, 136(12):4629–4640, 2008.
- P. J. van Leeuwen. Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Quarterly Journal of the Royal Meteorological Society*, 136:1991–1999, October 2010.
- E. Vanden-Eijnden. Numerical techniques for multi-scale dynamical systems with stochastic effects. *Communications in Mathematical Sciences*, 1(2):385–391, 2003.

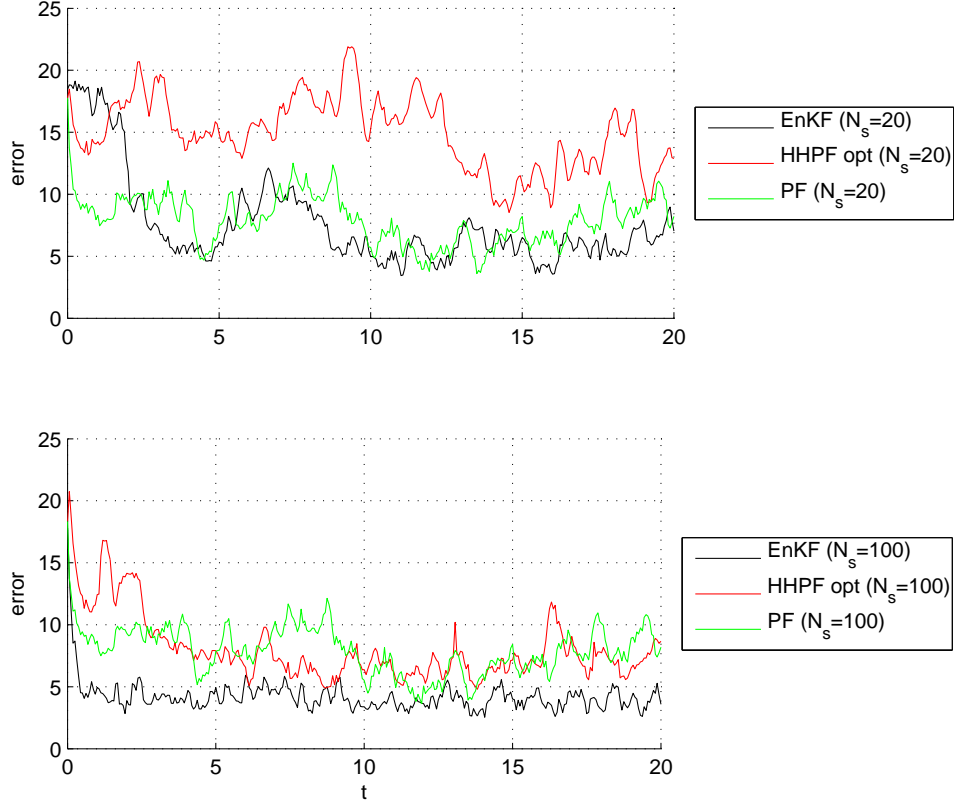


FIGURE 11. Sparse observations, comparison of estimation errors of the EnKF, PF and optimized HHPF with observation error. Estimation error of the optimized HHPF and the PF are of the same magnitude at $N_s = 100$. The optimized HHPF still required less computation time than the EnKF and PF even with the EnKF and PF being implemented using minimum N_s possible.

S. Wilks, Daniel. Effects of stochastic parametrizations in the lorenz '96 model. *Quarterly Journal of the Royal Meteorological Society*, 131(131):389–407, 2005.

M. Zakai. On the Optimal Filtering of Diffusion Processes. *Z. Wahrscheinlichkeitstheorie verw. Geb.*, 11(3): 230–243, 1969.

¹DEPARTMENT OF AEROSPACE ENGINEERING, UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN, 306 TALBOT LABORATORY, MC-236, 104 SOUTH WRIGHT STREET, URBANA, ILLINOIS 61801, USA
E-mail address: lingala1@illinois.edu, navam@illinois.edu, hyeong2@illinois.edu

²INSTITUT FÜR MATHEMATIK, HUMBOLDT-UNIVERSITÄT ZU BERLIN, RUDOWER CHAUSSEE 25, 12489 BERLIN, GERMANY
E-mail address: perkowsk@math.hu-berlin.de

